Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations | 1. Thesis and Dissertation Collection, all items

2020-12

DESIGN, BUILD AND TEST OF A LOW-COST, HIGH-BANDWIDTH X-BAND SOFTWARE-DEFINED RADIO FOR CUBESATS

West, Logan T.

Monterey, CA; Naval Postgraduate School

http://hdl.handle.net/10945/66741

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun

http://www.nps.edu/library

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

DESIGN, BUILD AND TEST OF A LOW-COST, HIGH-BANDWIDTH X-BAND SOFTWARE-DEFINED RADIO FOR CUBESATS

by

Logan T. West

December 2020

Thesis Advisor: Wenschel D. Lan
Co-Advisor: James H. Newman

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** December 2020	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE DESIGN, BUILD AND TEST OF A LOW-COST, HIGH-BANDWIDTH X-BAND SOFTWARE-DEFINED RADIO FOR CUBESATS	**5. FUNDING NUMBERS** RGNNU
6. AUTHOR(S) Logan T. West	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) DoD Space	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.	**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

The objective of this research was to design, build, and test the next iteration of a low-cost, high-bandwidth X-band software-defined radio (SDR). The flight version of this iteration of the payload will be integrated into a commercially provided 6U bus to transmit and receive data, commands, and telemetry between the satellite and the MC3 network. The spacecraft reference design included two other payloads and used the Astro Digital Corvus-6 bus as the baseline for defining payload-bus-ground interfaces. This project utilized MATLAB Simulink to program the SDR. The SDR will primarily operate in the store-and-forward mode for transmissions when in line-of-sight of a ground station. The up/down convert board was designed, and manufacturing options were explored. Additionally, this project finalized the mechanical enclosure and bus interfaces. The payload aimed to maintain a 0.5U CubeSat form-factor and achieve a data rate of up to 10 Mbps with 1e-5 bit error rate. Following design and construction, the hardware and software components were subjected to functional end-to-end testing to evaluate performance. Further flight qualification will subject the payload to environmental testing to ensure survivability through launch and the expected low-Earth orbit environment.

14. SUBJECT TERMS software-defined radio, SDR, X-band, CubeSat, Cube Satellite, commercial off-the-shelf, COTS	**15. NUMBER OF PAGES** 157
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89) Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

DESIGN, BUILD AND TEST OF A LOW-COST, HIGH-BANDWIDTH X-BAND SOFTWARE-DEFINED RADIO FOR CUBESATS

Logan T. West
Lieutenant, United States Navy
BS, U.S. Naval Academy, 2012

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ASTRONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 2020

Approved by: Wenschel D. Lan
Advisor

James H. Newman
Co-Advisor

Garth V. Hobson
Chair, Department of Mechanical and Aerospace Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The objective of this research was to design, build, and test the next iteration of a low-cost, high-bandwidth X-band software-defined radio (SDR). The flight version of this iteration of the payload will be integrated into a commercially provided 6U bus to transmit and receive data, commands, and telemetry between the satellite and the MC3 network. The spacecraft reference design included two other payloads and used the Astro Digital Corvus-6 bus as the baseline for defining payload-bus-ground interfaces. This project utilized MATLAB Simulink to program the SDR. The SDR will primarily operate in the store-and-forward mode for transmissions when in line-of-sight of a ground station. The up/down convert board was designed, and manufacturing options were explored. Additionally, this project finalized the mechanical enclosure and bus interfaces. The payload aimed to maintain a 0.5U CubeSat form-factor and achieve a data rate of up to 10 Mbps with 1e-5 bit error rate. Following design and construction, the hardware and software components were subjected to functional end-to-end testing to evaluate performance. Further flight qualification will subject the payload to environmental testing to ensure survivability through launch and the expected low-Earth orbit environment.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

8-PSK	eight-phase phase shift keying
A	ampere
ABC	Aft Bulkhead Carrier
ADC	analog-to-digital converter
AGC	automatic gain control
AMS	agile mixed signaling
AP	all programmable
ASIC	application specific integrated circuit
ASK	amplitude shift keying
BER	bit error rate
BPSK	binary phase shift keying
CCSDS	Consultative Committee for Space Data Systems
CDS	CubeSat Design Specification
CLB	configurable logic block
cm	centimeter
COTS	commercial-off-the-shelf
CSLI	CubeSat Launch Initiative
CubeSat	cube satellite
DAC	digital-to-analog converter
dB	decibel
DCS	digital communication system
DSP	digital signal processor
EDU	engineering development unit
EELV	evolved expendable launch vehicle
EIRP	equivalent isotropic radiated power
ESPA	EELV Secondary Payload Adapter
FCC	Federal Communications Commission
FEC	forward error correction
FFT	fast Fourier transform
FMC	FPGA Mezzanine Card

FPGA	field programmable gate array
FSBL	first stage boot loader
FSK	frequency shift keying
FVEY	The Five Eyes
g	gram
GB	gigabyte
GHz	gigahertz
gsg	ground-signal-ground
HAB	high-altitude balloon
HDL	hardware description language
HPA	high-power amplifier
IC	integrated circuit
IF	intermediate frequency
I/O	input/output
IDE	integrated design environment
ITU	International Telecommunication Union
JTAG	Joint Test Action Group
kg	kilogram
km	kilometer
LAB	low-altitude balloon
LDO	low-dropout regulator
LEO	low-Earth orbit
LNB	low-noise block
LO	local oscillator
LPC	low pin count
LV	launch vehicle
MarCO	Mars Cube One
MATLAB	Matrix Laboratory
MB	megabyte
Mbps	megabits per second
MC3	Mobile CubeSat Command and Control
MHz	megahertz

mm	millimeter
NASA	National Aeronautics and Space Administration
NLAS	Nanosatellite Launch Adapter System
NPS	Naval Postgraduate School
NPSCuL	NPS CubeSat Launcher
NTIA	National Telecommunications and Information Administration
NTM	national technical means
OTG	on-the-go
OQPSK	offset quadrature phase shift keying
PAE	power added efficiency
PCB	printed circuit board
PL	programmable logic
PLL	phase lock loop
P-POD	Poly Picosatellite Orbital Deployer
PS	processing system
PSK	phase shift keying
QPSK	quadrature phase shift keying
QSPI	quad serial peripheral interface
R&D	research and development
RCV	receive
RF	radio frequency
RFA	Radio Frequency Authorization
RTOS	real-time operating system
SDR	software-defined radio
SLO	San Luis Obispo
SMA	SubMiniature version A
SmallSat	small satellite
SNR	signal-to-noise ratio
SoC	system on a chip
SoM	system on a module
SSAG	Space Systems Academic Group
SSO	Sun-synchronous orbit

TIC	Terahertz Imaging Camera
TVAC	thermal vacuum chamber
UART	Universal Asynchronous Receiver/Transmitter
UDP	User Datagram Protocol
V	volts
VCO	voltage-controlled oscillator
VPN	virtual private network
W	watt
X-MW	X-Microwave
XMT	transmit
XSA	Xilinx Shell Archive

ACKNOWLEDGMENTS

I am forever indebted to the SSAG and the Small Satellite Laboratory staff for their patience and assistance while supporting my thesis and the X-band SDR project. Thank you to Mr. David Rigmaiden for your tireless effort helping me through communications theory, application, hardware implementation and interfacing. Thank you to Mr. Jim Horning for your continuous aid in software development and employment. Thank you to Mr. Dan Sakoda for drafting the mechanical enclosure for this payload. Most importantly, I would like to extend my utmost gratitude to my advisors and professors, Dr. Wenschel Lan and Dr. Jim Newman. Without your guidance and dedication, none of this would have been possible.

I would also like to thank my family and friends for always being available and ready to listen. Your love and support gave me strength throughout my time at NPS and pushed me to keep going during some of my most challenging moments.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. OBJECTIVE

The objective of this thesis is to design, build and test the next iteration of a low-cost, high-bandwidth X-band software-defined radio (SDR). The X-band SDR engineering development unit (EDU) will provide pivotal groundwork for a flight-like payload. The flight version of this iteration will be integrated into a commercially provided 6U bus to transmit and receive data, commands, telemetry and imagery between the bus with accompanying payloads and the Mobile CubeSat Command and Control (MC3) network. The spacecraft reference design will include two other payloads and will use the Astro Digital Corvus-6 bus as the baseline to define payload-bus-ground interfaces. The Corvus-6 bus features a 3U payload space and multiple data interfaces as shown in Figure 1 [1].

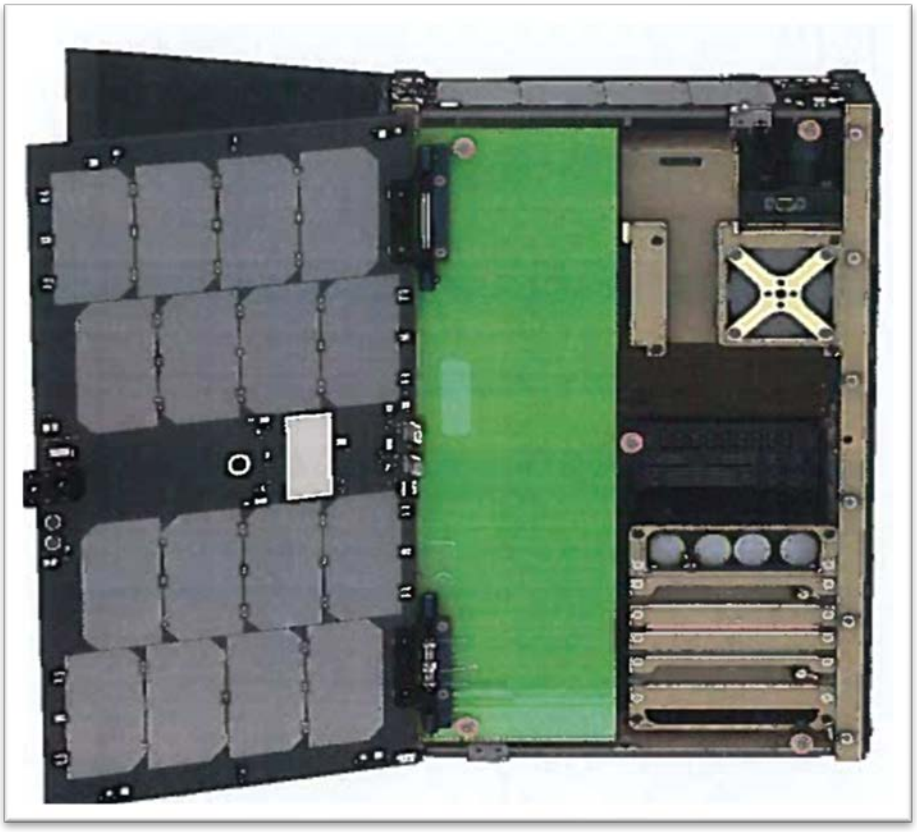


Figure 1. Corvus-6 bus with 3U payload space in green. Source: [1].

Ultimately, this research will determine the possibility of building an X-band SDR utilizing commercial off-the-shelf (COTS) components, if high data-rate transmission can be achieved in the X-band using COTS hardware and software, and demonstrate the feasibility to reconfigure software on-orbit.

Although CubeSats integrating SDR technology have been successfully demonstrated on-orbit, X-band SDR capabilities employing COTS components have yet to be flight-proven. A high-bandwidth on-orbit X-band SDR will significantly enhance the communications capability of CubeSats by employing frequencies and modulations of expanding relevance. The utilization of low-cost, COTS hardware will allow for rapid prototyping and development while the SDR itself will afford cross-functionality and adaptability during operations. The flight iteration of this project will demonstrate, through pathfinding and on-orbit activities, advantages in an increasingly contested space environment.

B. CUBESATS

Cube satellites, or CubeSats, are a classification of small spacecraft intended to reduce cost by adopting a standardized size and form factor defined by the unit "U." Originally developed in 1999 through a collaborative effort between professors at California Polytechnic State University (Cal Poly), San Luis Obispo (SLO) and Stanford University, the CubeSat project proposed reduced cost and development time for small satellites (SmallSats) while increasing accessibility to space through sustained frequent launches [2]. While a SmallSat is generally defined as any satellite weighing less than 300 kg, a CubeSat further refines the shape, size and weight to a 10 cm cube (10 cm x 10 cm x 10 cm) reference unit with a mass of approximately 1 to 1.33 kg generally known as 1U [3]. Specifically, a CubeSat is categorized as a nanosatellite, maintaining a mass between 1 to 10 kg and 1U to 12U reference size. These standards are maintained in the CubeSat Design Specification (CDS), developed and revised by The CubeSat Program at Cal Poly SLO [2]. The classification for SmallSats is shown in Table 1.

Table 1. Classes of small satellites. Source: [4].

Class	Mass (kg)
Minisatellite	100 to 180
Microsatellite	10 to 100
Nanosatellite	1 to 10
Picosatellite	0.01 to 0.1
Femtosatellite	0.001 to 0.01

Since its inception, the CubeSat has evolved to the common standard sizes of 1U, 1.5U, 2U, 3U, 6U, and 12U, as shown in Figure 2.

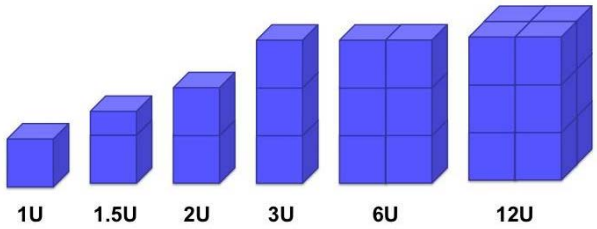


Figure 2. CubeSat standard sizes. Source: [4].

Unlike traditional large satellites which can be cost and schedule prohibitive, CubeSats provide increased gap filling capability for government, academic, and industry applications. CubeSats offer a cost-effective platform through their use of mass produced, COTS components and standardized form-factor, in conjunction with their afforded reduced testing standards and rapid deployment. Programs like NASA's CubeSat Launch Initiative (CSLI) facilitate technology partnerships, providing the opportunity for academic organizations to conduct low-cost research and development (R&D) efforts and gain hands-on experience with developing flight hardware [5].

CubeSats are customarily launched as secondary payloads and interface with a launch vehicle (LV) through a CubeSat dispenser. Although ride-sharing saves on cost for CubeSats to get into orbit, this subjects the satellite to higher environmental stress during launch and may dispense the CubeSat in a less than ideal orbit. Some prominent examples

of CubeSat dispenser systems include the Poly Picosatellite Orbital Deployer (P-POD) and Tyvak 6U Nanosatellite Launch Adapter System (NLAS), which successfully dispensed the Mars Cube One (MarCO) A and B twin communications-relay CubeSats as they flew past Mars [6]. Similarly, systems like the Aft Bulkhead Carrier (ABC) provide simple and inexpensive adapters to attach multiple P-PODs to a single evolved expendable launch vehicles (EELV) Secondary Payload Adapter (ESPA), therefore greatly increasing rideshare opportunities for CubeSats [7]. The Department of Defense (DOD) is seeking to implement CubeSats as an alternative to larger satellites because they provide a low-cost alternative for the rapid deployment of short-term space-based capabilities.

C. SOFTWARE-DEFINED RADIOS

The application of SDRs is becoming more prevalent in CubeSat and small satellite communications because of their ability to reconfigure parameters through software. Traditional radios are hardware defined, limiting modification and cross-functionality, especially on-orbit. In these systems, changes are employed by physical alteration of the radio hardware. Unlike these hardware-based platforms, SDRs use features such as field programmable gate arrays (FPGAs), a system on a chip (SoC), and digital signal processors (DSPs) to provide multiple modes, bands and functionality to support different waveforms through software changes. SDRs allow the user to incorporate data processing and modulation into programmable logic (PL) and software, so that changes can be made easily via updates. For space applications, SDRs are desirable because they can make software adjustments to reconfigure during operations to support various mission objectives without adding or changing hardware. Moreover, SDRs replace traditional hardware-based components with microcontrollers and integrated circuits (ICs), allowing for a reduced form-factor to a size more easily integrated into CubeSat sized spacecraft.

D. MOBILE CUBESAT COMMAND AND CONTROL (MC3) NETWORK

The MC3 ground station network is a DOD effort “to build common-use infrastructure supporting communications and mission operations of small satellites for a wide range of U.S. government organizations, contractors, universities, and foreign partners” [8]. Currently operated by the NPS Space Systems Academic Group (SSAG), the

MC3 network seeks to provide bent-pipe access to satellite users over an internet connection through "low-cost ground station terminals" at contributing academic and government institutions [8]. Presently, there are eight ground station locations and three international participants, summarized in Table 2.

Table 2. MC3 station locations. Adapted from [8].

Site (Designator)	Location	Capability
Hawaii Spaceflight Laboratory (HSFL)	Honolulu, HI	UHF
Naval Postgraduate School (NPS)	Monterey, CA	UHF/S-band
Space Dynamics Laboratory (SDL)	Logan, UT	UHF/S-band
University of New Mexico/ Cosmiac (UNM)	Albuquerque, NM	UHF/S-band
Air Force Institute of Technology (AFIT)	Dayton, OH	UHF/S-band
US Coast Guard Academy (USCGA)	New London, CT	S-band
Malabar Transmitter Annex (MLB)	Palm Bay, FL	UHF/S-band
University of Alaska, Fairbanks (UAF)	Fairbanks, AK	S-band

MC3 ground stations are "networked together with COTS virtual private network (VPN) devices for secure, cost-effective communications between sites" allowing satellite operators to log in remotely via VPN to control their spacecraft [8]. Figure 3 shows ground station locations and satellite orbital parameters from the MC3 network terminal at NPS.

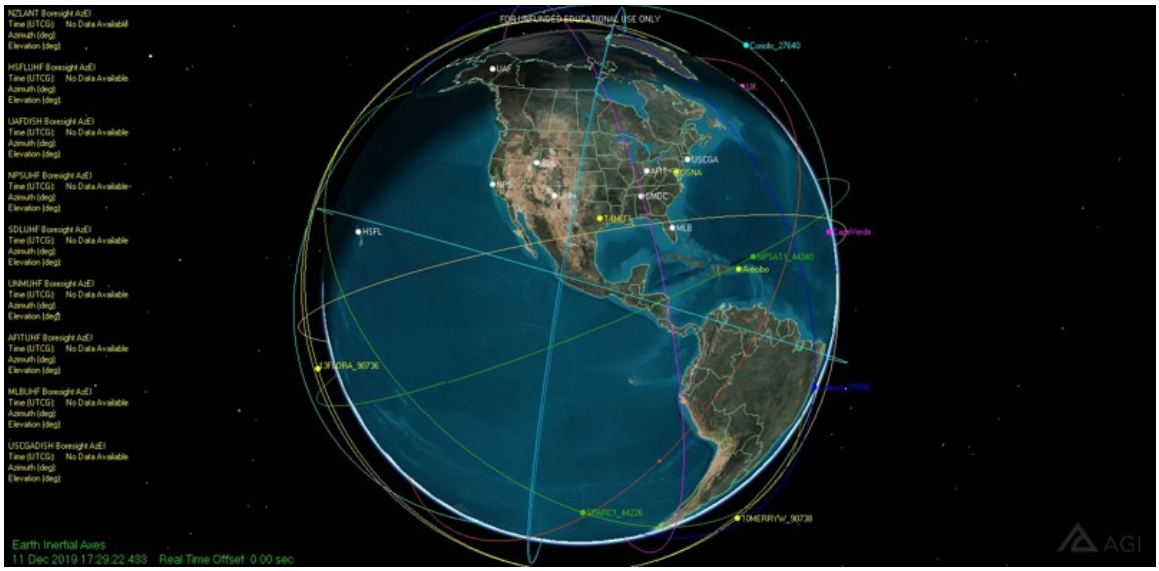


Figure 3. MC3 ground station overview

COTS hardware is utilized where possible while government-owned, open-source software is used for data routing and DSP through SDRs. Commercial SDRs, such as the National Instruments USRP-2922 and Kratos quantumRadio, are being implemented at MC3 ground stations to capitalize on cross-compatibility and cater to a broader range of users [9]. Although MC3 currently operates in the UHF and S-band frequencies, this research will develop the implementation and testing of X-band capability to validate technologies for both ground and space applications. Table 3 shows the current and planned frequency ranges of the MC3 network.

Table 3. MC3 frequency ranges. Adapted from [8].

Band	Frequency	Designator
UHF uplink	449.75-450.25 MHz	12K5F1D 43K0F1D
UHF downlink	902-928 MHz	115KG1D
S-band uplink	2025-2110 MHz	2M00G2D 2M45G1D
S-band downlink	2200-2290 MHz	1M60G1D 2M00G2D 2M45G1D
X-band uplink	7190-7250 MHz	(future)
X-band downlink	9025-8400 MHz	(future)

E. RADIO FREQUENCY

The radio spectrum is the radio frequency (RF) component of the overarching electromagnetic spectrum [10]. Within the United States, the responsibility for radio spectrum allocation and regulation is separated between the Federal Communications Commission (FCC), which monitors and administers spectrum for non-federal applications, and the National Telecommunications and Information Administration (NTIA), which controls spectrum for federal applications [10]. These organizations regulate spectrum management between 9 kHz and 275 GHz through the United States Table of Frequency Allocations [11]. Federal requests for spectrum allocation are routed through the NTIA utilizing a Certificate of Spectrum Support and Radio Frequency Authorization (RFA). For the international component, the International Telecommunication Union (ITU), which is a specialized agency of the United Nations, manages global spectrum and satellite orbits while "developing technical standards to ensure networks interconnect seamlessly" [12].

The X-band spectrum, which includes frequencies between 8 and 12 GHz, is advantageous because it allows for higher transmission data-rate and more bandwidth than its L, S, and C-band spectrum counterparts. For example, the Innoflight Compact L/S-band radio (SCR-104) provides a maximum data-rate of 4.5 Mbps while the Innoflight Compact X-band Transceiver (SCR-106) offers up to 132 Mbps [13], [14]. High data-rate transmission requires more bandwidth. In order to achieve more bandwidth, a higher carrier frequency, such as X-band, becomes necessary. This high-data transmission rate, or throughput, is a feature which the MC3 currently lacks, but will be required to support high-data capacity payloads like the Terahertz Imaging Camera (TIC) [15]. Furthermore, X-band is less crowded for satellite communications than other spectrums. Data transmission between ground stations and on-orbit spacecraft has become increasingly congested on lower bands, such as S and C-bands, because they have been more readily available for CubeSat applications in the past. Additionally, X-band provides resiliency against environmental effects such as rain fade which degrades RF signals at frequencies above 11 GHz. The frequency range of 8025–8400 MHz was used for this research to design, build, and test hardware and software components of the X-band SDR payload.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND

A. DIGITAL COMMUNICATIONS

Unlike analog communications systems, which send a waveform from an infinite variety of waveform shapes with infinite possible resolution, a digital communication system (DCS) sends a waveform from a finite set of possible waveforms in a finite time interval [16]. Digital systems offer flexibility, stability, reliability, and noise resilience over analog systems. The primary advantage of a digital system is its ability to regenerate a signal after distortion of the waveform from transmission lines, unwanted electrical noise, and other interferences [16]. Transmission in digital communications starts with an analog information source which is then sampled and encoded to a digital input known as a bit stream, represented by the binary states of either 0 or 1 (off and on, respectively). This two-state operation "facilitates signal regeneration and thus prevents noise and other disturbances from accumulating in the transmission" [16]. Before this bit stream can be transmitted, it must be transformed to a digital waveform, through a process known as modulation. For the receive component, this process is reversed so that the waveform is demodulated to its original binary form. Figure 4 portrays a typical block diagram for a DCS where the upper blocks represent the transmit (XMT) component and the lower blocks represent the receive component (RCV).

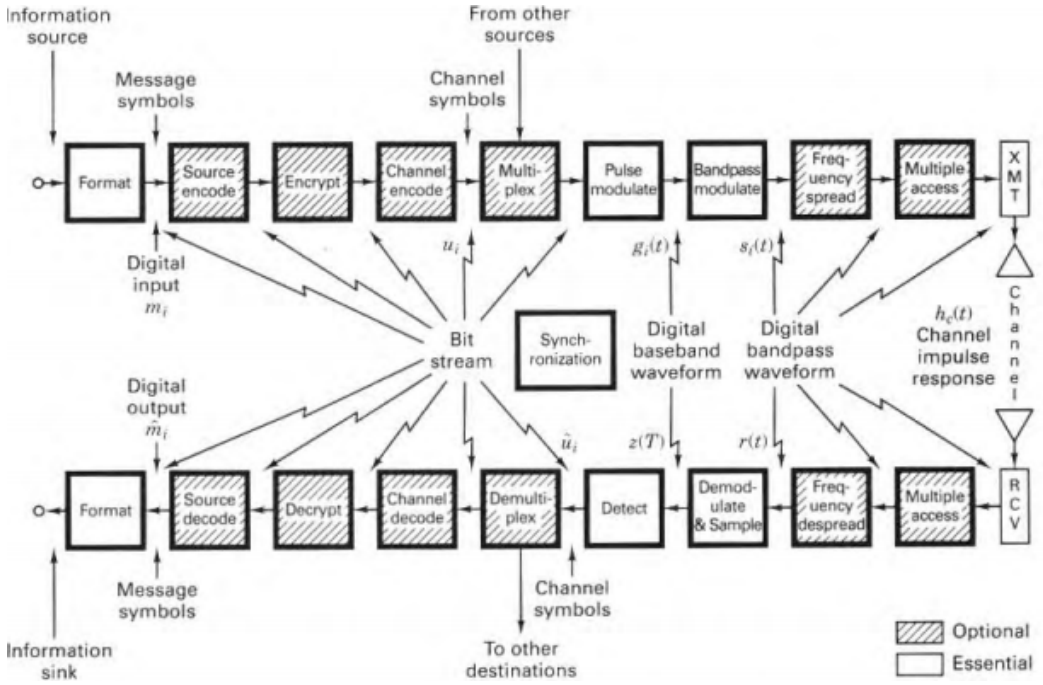


Figure 4. Block diagram for a typical DCS. Source: [16].

Modulation is the procedure "by which message symbols are converted to waveforms that are compatible with the requirements imposed by the transmission channel" [16]. In this signal processing practice, keying is applied to the amplitude, frequency, or phase of a carrier signal in congruence with information from the binary digits [17]. The most common modulation methods include amplitude shift keying (ASK), frequency shift keying (FSK), and phase shift keying (PSK). While each scheme has its advantages and disadvantages, PSK is preferred for satellite communications because it provides increased protection against noise. PSK contains signal information in its phase while noise mainly affects the amplitude of the carrier [17]. Therefore, the most prominent modulation schemes utilized in satellite communications are binary PSK (BPSK), quadrature PSK (QPSK) or offset QPSK (OQPSK), and eight-phase PSK (8-PSK) [17].

This research focused on the QPSK modulation scheme as it is the favored method of modulation for X-band satellite communications. For QPSK modulation, the phase of the carrier takes on one of M possible values or symbols, where in the case of QPSK M = 4. Each symbol is a carrier frequency sinusoid having one of M possible phases spaced $2\pi/M$ apart [17]. This 4-phase PSK scheme separates any two adjacent phasors by 90° with

the phase of the carrier taking on one of four values: 45°, 135°, 225°, or 315°, as shown in Figure 5 [17].

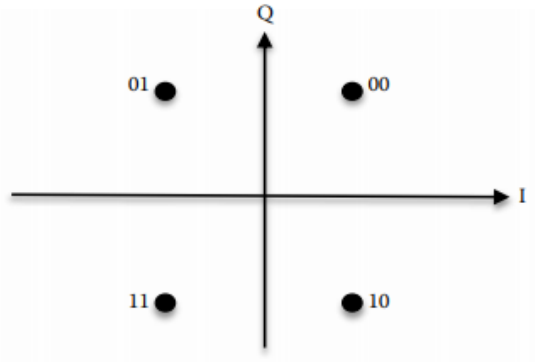


Figure 5. QPSK constellation diagram. Source: [17].

A QPSK modulator divides an incoming bit stream into an I- (in-phase) channel and a Q- (in-quadrature) channel. The data from these channels are fed to balanced product modulators and then summed bit-wise to a single output data stream [17]. Figure 6 depicts a QPSK block diagram for this process. Figure 7 shows the resulting waveform from the summation of a normal I and normal Q, representing in a phase shift of 45° [18].

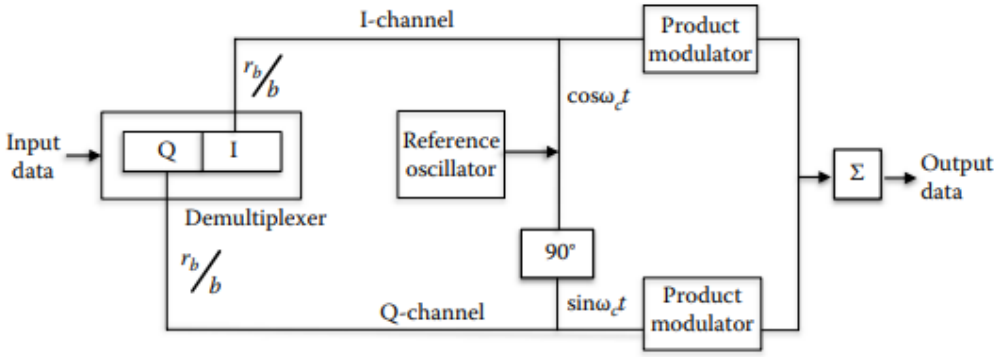


Figure 6. QPSK modulator block diagram. Source: [17].

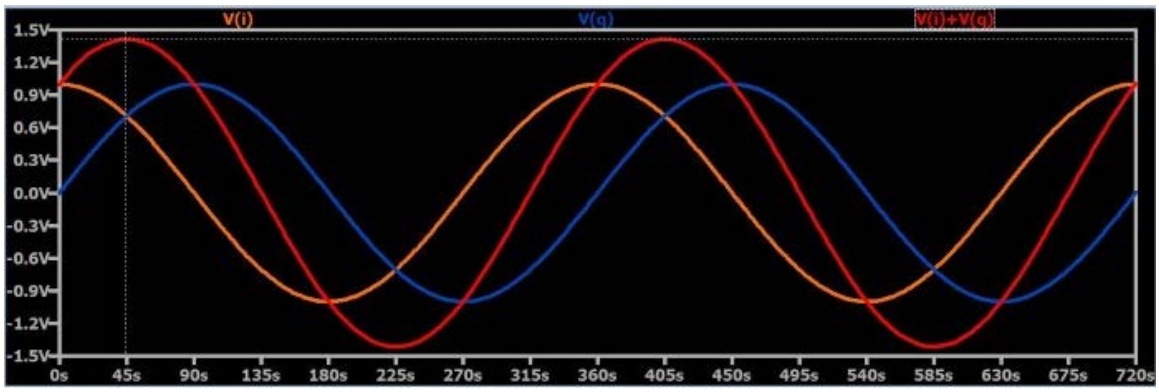


Figure 7. I normal and Q normal summation. Source: [18].

B. SOFTWARE-DEFINED RADIO

The term SDR refers to a radio system in which the majority of the functionality related to the physical layer is executed in software utilizing DSP algorithms [19]. Tuttlebee defines an SDR as any radio "in which the receive digitization is performed at some stage downstream from the antenna, typically after wideband filtering, low noise amplification, and down conversion to a frequency in subsequent stages – with a reverse process occurring for the transmit digitization" [20]. The characteristics of the radio are defined in digital signal processing contained in flexible, reconfigurable functional blocks. As SDR technology advances, digitization will be done at or very near the antenna, while all processing for the radio will be done through software residing on fast DSP elements [20]. Figure 8 shows the block diagram for a modern SDR transceiver.

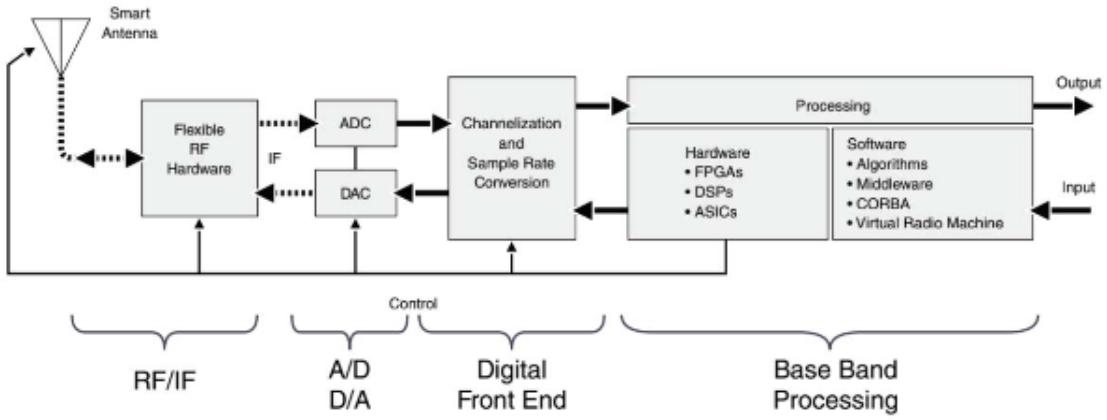


Figure 8. SDR block diagram. Source: [21].

The primary elements of an SDR are the RF front end, analog-to-digital converter (ADC), digital-to-analog (DAC), digital front end, and FPGA. The RF front end, which includes the flexible RF hardware near the antenna, converts the analog source to its IF on the receive side so that it is compatible with the ADC. The DAC provides data conversion on the transmit side to push the IF through the RF back end, which includes all baseband processing. The digital front end serves as the bridge between RF and baseband processing, by performing the functionalities of channelization (down-conversion from RF to baseband and filtering), digitization, sample-rate conversion, and synchronization [20]. Baseband processing is driven by the FPGA, DSP or application specific integrated circuit (ASIC). The DSP is an optimized general-purpose microprocessor while the FPGA is a low-cost alternative to the design of ASICs [20]. A typical FPGA consists of an array of configurable logic blocks (CLBs) surrounded by configurable routing, which allows the FPGA to execute parallel and pipelined data flow for high-speed signal processing [20].

C. COMMERCIALLY AVAILABLE X-BAND RADIOS

Although commercial options provide X-band SDR functionality in a CubeSat sized form factor, these prebuilt systems are prohibitively expensive, require sustained manufacturer support to operate and modify, and may be incompatible with the MC3 architecture. This ultimately inhibits the end user on the MC3 network from optimally building and maintaining a high-bandwidth communication network.

One such example is the Innoflight Compact X-band Transceiver (SCR-106). This transceiver comes in both CubeSat and SmallSat variant enclosures, provides a transmit frequency range of 7900 MHz to 8500 MHz, and a transmit data rate of up to 132 Mbps [14]. While this transceiver provides all of the desired capability in a fully tested, flight-proven design, the SCR-106 retails for $120,000 per unit, not including the cost of high-power amplifiers, diplexers, or splitters. Additionally, the manufacturer requires mandatory Communications Systems Engineering Support, adding $20,900 per order with a standard lead time of six months. Another example is the IQ Spacecom X Band Transceiver SDR [22]. Both of these commercial transceiver options provide the ability to transmit and receive.

An example of a transmit only commercial X-band system is the Syrlinks EWC27, which provides data rates of up to 100 Mbps in the 8025 MHz to 8400 MHz frequency range at a cost of $89,900/per unit. EnduroSat's X-band Transmitter provides up to 150 Mbps in this same frequency range at a slightly more appealing price point of $26,800. Another commercial X-band SDR transmitter in the CubeSat form factor is the GomSpace NanoCom XT8250 [23]. Although these commercial options provide the X-band SDR capability sought by MC3, they are expensive platforms which tie the user to the manufacturer, and work against the low-cost network and common use infrastructure that MC3 is attempting to implement.

D. LINK BUDGET

A satellite link consists of two parts: the uplink transmits from a ground station to the satellite, while the downlink transmits from the satellite to a ground station [17]. The link encompasses the full communications path and conducting link analysis allows us to estimate performance of a communication system. In order to close the link and ensure the signal is detected by the receiver, there must be a positive link margin. The most important criteria in a satellite link is signal quality, defined by the ratio of energy per bit (E_b) and noise density (N_0) and otherwise utilized in link analysis as the normalized signal-to-noise ratio (SNR) [16]. The equation to calculate $\frac{E_b}{N_0}$ is shown in Equation (1)

$$\frac{E_b}{N_0} = \frac{C}{N_0} - R_b \,. \tag{1}$$

Here, $\frac{C}{N_0}$ is the carrier power to noise ratio while R_b is the data rate. $\frac{E_b}{N_0}$ is utilized to determine the bit error rate (BER) and ultimately the link margin, *M*.

The basic link budget equation allows for the calculation of gains and losses for a signal as shown in Equation (2) [24]

$$P_{Rx} = P_{Tx} + G_{Tx} + G_{Rx} + L_{Tx+Rx} \,. \tag{2}$$

In this equation, P_{Rx} is received signal power, P_{Tx} is transmitted signal power, G_{Tx} is the gain of the antenna on the transmit side, G_{Rx} is gain on of the antenna on the receive side, and L_{Tx+Rx} is total losses to include transmit loss, receiver loss, atmospheric losses, and free space dispersion losses [16]. These elements are expressed in decibels (dB) and add if they increase the received signal or subtract if they decrease the received signal. Antenna gain is a function of the antenna efficiency, shape, size, and signal frequency. The equation for determining antenna gain is shown in Equation (3) [24]

$$G = 20.4 + 20\log(f) + 20\log(D) + 20\log(\eta) \,. \tag{3}$$

Here, f is the signal frequency in GHz, D is antenna diameter in meters, and η is a unitless measure of antenna efficiency. Free space loss is calculated as shown in Equation (4) [24]

$$L_s = 92.45 + 20\log(S) + 20\log(f) \,. \tag{4}$$

In this equation, L_S is free space path loss in dB, S is the separation distance between the transmit antenna and receive antenna in km, and f is the signal frequency in GHz. Equivalent isotropic radiated power (EIRP) is the summation of transmitted signal power, P_{Tx}, effective transmit antenna gain, G_{Tx}, and loss in the transmit cable, L_c, shown in Equation (5) [24]

$$EIRP = P_{Tx} + G_{Tx} + L_c \,. \tag{5}$$

The link margin equation is defined as the difference between the achieved $\frac{E_b}{N_0}$ and the required $\frac{E_b}{N_0}$, where *M* is the link margin in dB as shown in Equation (6) [24]

$$M = \left(\frac{E_b}{N_0}\right)_{achieved} - \left(\frac{E_b}{N_0}\right)_{required} . \quad (6)$$

For this research, a link analysis was conducted utilizing the X-band SDR system parameters and amplifying assumptions for a worst case circular low-Earth orbit (LEO), Sun-synchronous orbit (SSO) environment at an altitude of 1200 km and elevation angle of 10°. Assumptions for additional parameters, such as BER and data rate, were based upon the final tested results from the previous design iteration [25]. Table 4 shows the summary of results for this link budget analysis with the excel spreadsheet used for calculations shown in Appendix A.

Table 4. Link budget analysis worst case.

Parameter	**Magnitude**	**Units**
Frequency	8.2125	GHz
Elevation angle	10	degrees
Altitude	1200	km
Data rate	1.2	Mbps
Tx losses	-1	dB
Rx losses	-1	dB
Implementation loss	-1.5	dB
Transmit power: 4W	6.021	dbW
Tx antenna gain	4	dB
EIRP	9.021	dBW
Free space loss	-180.65	dB
Antenna G/T	23.40	dB/K
BER	1.5e-5	N/a
Achieved E_b/N_0	17.08	dB
Required E_b/N_0	9.65	dB
Link Margin	**7.43**	**dB**

An additional link analysis was conducted using reasonable LEO parameters at an altitude of 500 km. The transmission rate for this analysis was increased to 10 Mbps as this

data rate is more realistic for an X-band SDR payload. Table 5 shows the summary of these results with the excel spreadsheet used for calculations seen in Appendix A. In both analyses, the link margin is sufficient to ensure the signal is detected by the receiver. Neither link analysis included forward error correction (FEC), which would add coding gain and improve the link margin.

Table 5. Link budget analysis reasonable case.

Parameter	**Magnitude**	**Units**
Frequency	8.2125	GHz
Elevation angle	10	degrees
Altitude	500	km
Data rate	10	Mbps
Tx losses	-1	dB
Rx losses	-1	dB
Implementation loss	-1.5	dB
Transmit power: 4W	6.021	dbW
Tx antenna gain	4	dB
EIRP	9.021	dBW
Free space loss	-175.32	dB
Antenna G/T	23.40	dB/K
BER	1.5e-5	N/a
Achieved E_b/N_0	13.21	dB
Required E_b/N_0	9.65	dB
Link Margin	**3.56**	**dB**

E. RELEVANT NPS RESEARCH

Several theses and projects conducted through the SSAG have preceded this research project, and have included topics such as: CubeSat and SmallSat development, digital communications systems development utilizing SDRs, flight testing utilizing high-altitude balloons (HABs), and ground station development and implementation through the MC3 network.

For example, Lovdahl developed the Com-Cube in 2018 to demonstrate the transmission of digital data utilizing an SDR as a CubeSat payload [26]. The Com-Cube

operated in the C-band frequency range and was flight tested on a low-altitude balloon (LAB) [26]. The payload successfully transmitted pictures from a wide-angle Raspberry Pi camera back to a ground station terminal in the field [26]. The Com-Cube was operated by an on-board Raspberry Pi 3 Model B and successfully demonstrated the ability to transmit and receive using an Ettus B205mini-i SDR [26]. The software chosen to run the SDR was open source software (GNU Radio) [26].

Bischoff continued the development of an X-band SDR as a payload as a 1.1U payload on a HAB within a 2U CubeSat form-factor bus [27]. For this proof-of-concept effort, GNU Radio was again utilized to run an Ettus B205mini-i SDR [27]. In this iteration, a single circuit-board Raspberry Pi 3 host computer controlled the SDR, initialized the frequency synthesizer, interfaced with the HAB bus and capture, and stored and processed data for transmission [27]. Bischoff identified the use of the Raspberry Pi 3 to control the SDR as a limiting factor for achieving a max data transmission rate of 1 Mbps [27]. Figure 9 depicts the block diagram of this X-band SDR iteration configured for 10 GHz [27]. Figure 10 shows the constructed X-band SDR payload inside the 2U HAB bus.

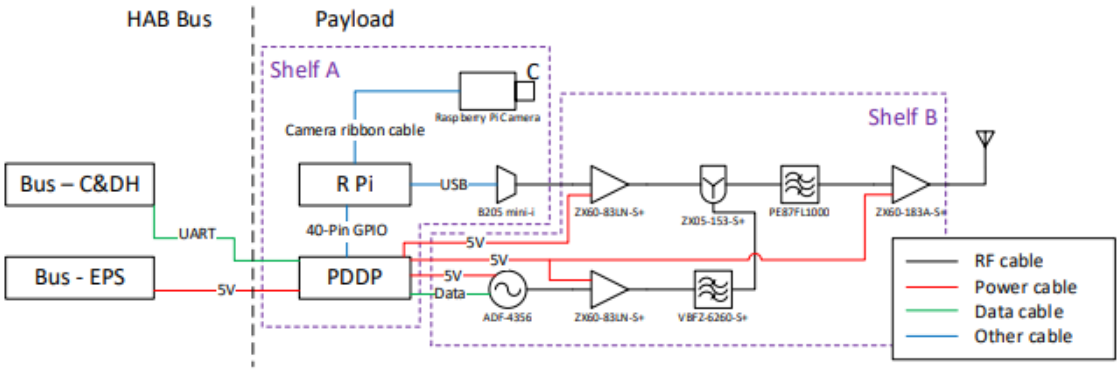


Figure 9. Block diagram of X-band SDR payload. Source: [27].

Figure 10. 1.1U CubeSat X-band SDR. Source [27].

The NPS payload design course (SS3861/SS4861) further refined this payload using COTS, drop-in RF parts to replace the larger RF components [25]. Additionally, the Ettus B205mini-i was replaced with the Analog Devices AD-FMComms3-EBZ RF SDR [25]. During this effort, the AVNET ZedBoard Software Development Kit replaced the Raspberry Pi 3 computer, and utilized MATLAB Simulink instead of GNU Radio to run the SDR [25]. The SDR in this design iteration was connected to the development board via the FPGA Mezzanine Card (FMC) low pin count (LPC) port. A separate host computer was required to run the Communications Toolbox Support Package for the Xilinx Zynq-Based Radio, in order to write the SDR card to initialize the ZedBoard as well as to run the MATLAB Simulink SDR mode [25]. The modulation scheme for this design iteration was BPSK. Functional and end-to-end testing conducted on the system showed an increase of data transmission to 1.2 Mbps [25]. Environmental end-to-end testing in the NPS thermal vacuum chamber (TVAC) subjected the drop-in RF components to a 17-hour test profile which validated functionality during low, high, and envelop-expanding duty cycles. Figure 11 shows the AD-FMComms3-EBZ SDR mounted onto the ZedBoard through the FMC-LPC port.

Figure 11. AVNET ZedBoard, AD-FMCOMMS3-EBZ RF SDR. Source [28].

Bower utilized several different SDR platforms to develop and evaluate custom MATLAB Simulink SDR models, which would test the interoperability of hardware/software combinations with MC3 ground receivers [29]. BPSK and QPSK modulation schemes were tested and validated using the SpectralNet digitizer and Kratos quantumRadio software modem [29]. Bower successfully implemented Reed-Solomon channel coding, interleaving, and randomization with BPSK modulation [29]. Use of SDR hardware as a peripheral rather than as a real-time operating system (RTOS) hindered data transmission rate [29]. Similarly, data transmission rate was slowed by the use of the SDR to perform all coding, interleaving, and randomization rather than dedicating these processes to be run by the SoC.

F. SYSTEM REQUIREMENTS

Several system requirements were established to meet the objectives of this research. The X-band SDR payload will be required to interface with the bus through ethernet and will be responsible for downlinking data gathered by the TIC to the MC3 network ground stations. In order to support this high data capacity payload, a requirement for this research was to achieve a data transmission rate of 1 Mbps or greater while using QPSK modulation. Another was to maintain a size of less than 0.5U and mass of less than

1 kg based on the size and weight limitations of radio assembly mechanical enclosure. Additionally, the implementation of COTS components where possible was required to reduce cost and shorten the development life cycle.

THIS PAGE INTENTIONALLY LEFT BLANK

III. HARDWARE

Hardware for this research was chosen in order to produce a cost-effective, COTS X-band SDR system that meet the desired objectives and requirements. To reduce the overall cost of the system, COTS hardware, such as drop-in RF prototyping components from X-Microwave, allow for the development and testing of X-band solutions that operate at intermediate frequencies (IF) in lower bands. Likewise, development kits like the ZedBoard allow for exploration of the Xilinx Zynq-7000 all programmable (AP) system on a chip (SoC), to manipulate payload-bus interfaces for rapid prototyping and proof-of-concept design. The initial hardware design for this research continued from the previous design iteration developed by the NPS payload design course [25].

A. INITIAL HARDWARE DESIGN

1. Development Board

The AVNET Zedboard was employed for the initial design because it is a low-cost development board, which incorporates the Xilinx Zynq-7000 AP SoC XC7Z020-CLG484-1 for its on-board processing [30]. The ZedBoard measures 16.0 by 13.5 cm and weighs approximately 0.166 kg. For memory, the ZedBoard includes 512 MB DDR3, 256 MB quad serial peripheral interface (QSPI) flash, and a 4 GB SD card. For communications ports, the board includes onboard USB Joint Test Action Group (JTAG) programming, 10/100/1000 ethernet, USB on-the-go (OTG) 2.0 and USB Universal Asynchronous Receiver/Transmitter (UART) [30]. Expansion connectors include an FMC-LPC connector, five Pmod headers, and an agile mixed signaling (AMS) header [30]. HDMI, VGA, and 128x32 OLED serve as the display outputs while eight LEDs, seven push buttons, and eight DIP switches makeup the general-purpose input/output (I/O) [30]. Figure 12 depicts the functional overlay of the ZedBoard. The hardware block diagram for the ZedBoard is shown in Figure 13.

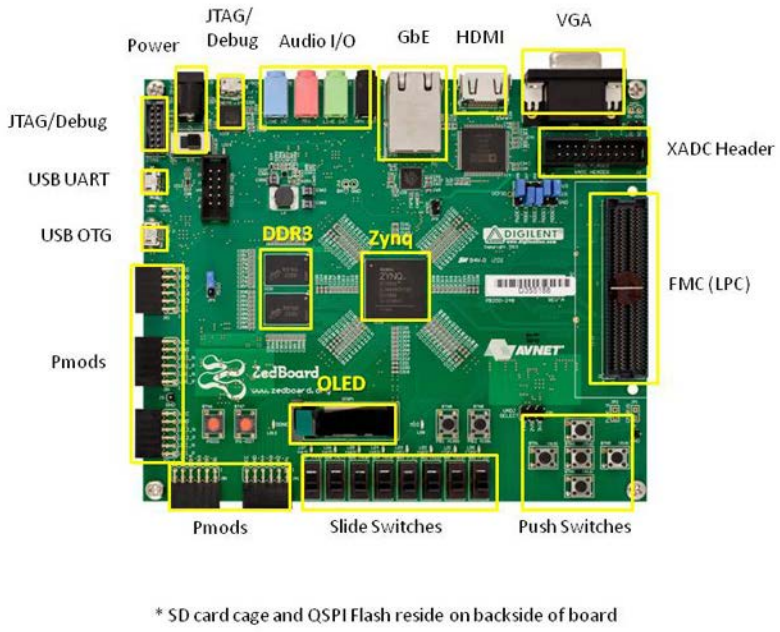


Figure 12. ZedBoard functional overlay. Source: [30].

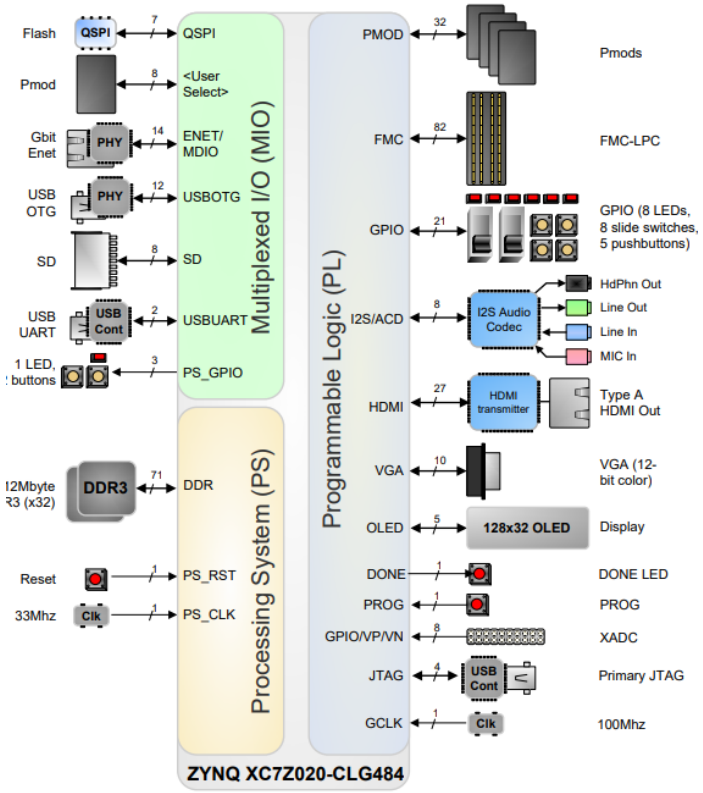


Figure 13. ZedBoard hardware block diagram. Source: [31].

The ZedBoard was chosen as the development board because it is low cost ($449), incorporates the powerful Zynq-7000 AP SoC, and provides several on-board peripherals and expansion capabilities for ease of use. While the ZedBoard offers an excellent prototyping environment for SDR development and testing, its size prevents potential integration into flight hardware.

2. SoC

The Xilinx Zynq-7000 AP SoC combines a dual core Cortex-A9 MPcore based processing system (PS) with 85,000 Series-7 Xilinx PL cells to provide high-performance, simplified embedded processor design in a single device [32]. The ZedBoard incorporates the Z7020 variant of the Zynq-7000 SoC which enables processing speeds up to 866 MHz, offers 220 DSP slices and 4.9 Mb of RAM [33]. This SoC offers the "flexibility and scalability of an FPGA while providing performance, power, and ease of use typically associated with ASICs" [32]. The SoC is broken down into the PS (application processor unit, memory interfaces, I/O peripherals, interconnect), and the PL. The primary boot configuration of the SoC is QSPI flash while the auxiliary boot configurations include cascaded JTAG and SD card [31]. While JTAG is considered the primary boot configuration for development and debugging, QSPI and SD card configurations are much faster and intended for final hardware/software deployment. The SoC allows for processing intensive operations, such as modulation/demodulation, to be performed by the programmable logic, while separate tasks like data decoding, user interface, and system monitoring are handled by the processing unit [34]. The block diagram for the Zynq-7000 SoC is shown in Figure 14.

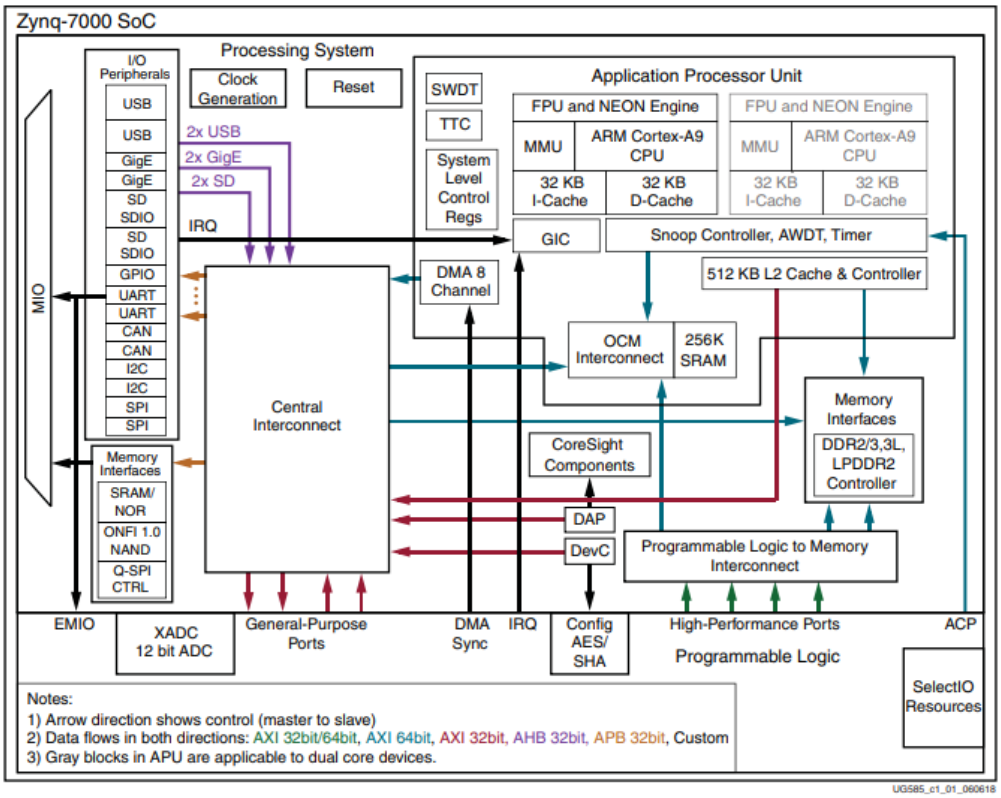


Figure 14. Zynq-700 SoC block diagram. Source: [32].

The Zynq-7000 AP SoC allows for RTOS to be deployed directly on the dual-core ARM processors to provide parallel processing, fast computation, and real-time performance necessary for a high-performance SDR system. Additional reference material for the Zynq-7000 AP SoC can be found in the *ZedBoard Hardware User's Guide* [31], *Zynq Technical Reference Manual* [32], and the Zynq-7000 SoC Data Sheet (Appendix B) [35].

3. SDR

The Analog Devices AD-FMComms3-EBZ 2 x 2 SDR rapid development and prototyping board provides SDR solutions which allowed the author to interface with the FPGA [34]. This SDR evaluation board measures 73.3 by 69 mm and weighs approximately 0.063 kg. The SDR provides two transmit and two receive ports through SubMiniature version A (SMA) coaxial RF connectors, allowing for full duplex communication in both directions simultaneously. This FMC module incorporates an

AD9361 integrated RF agile transceiver with a tuning range of 70 MHz to 6 GHz and supported bandwidth of 200 kHz to 56 MHz that connects easily to the Xilinx FPGA development platform [36]. The IC "combines an RF front end with a mixed-signal baseband section and integrated frequency synthesizers" [37]. The AD-FMComms3-EBZ SDR overview and AD9361 IC block diagram are shown in Figure 15 and Figure 16, respectively.

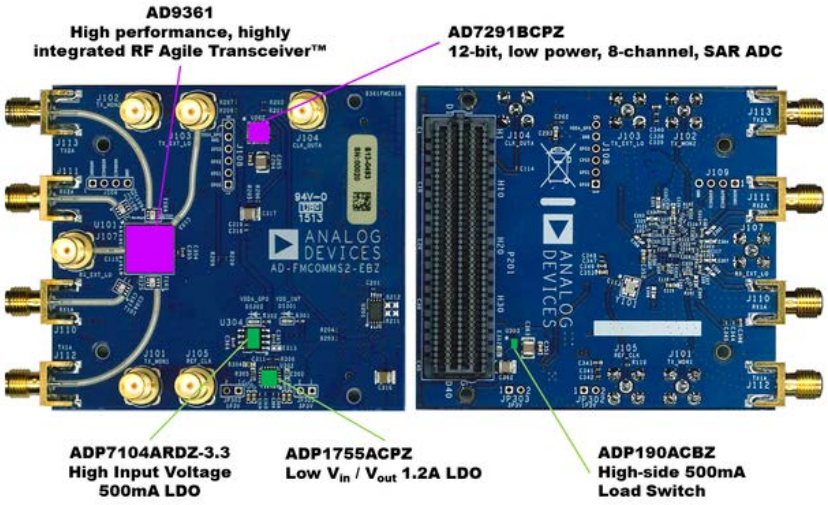


Figure 15. AD-FMComms3-EBZ SDR. Source [36].

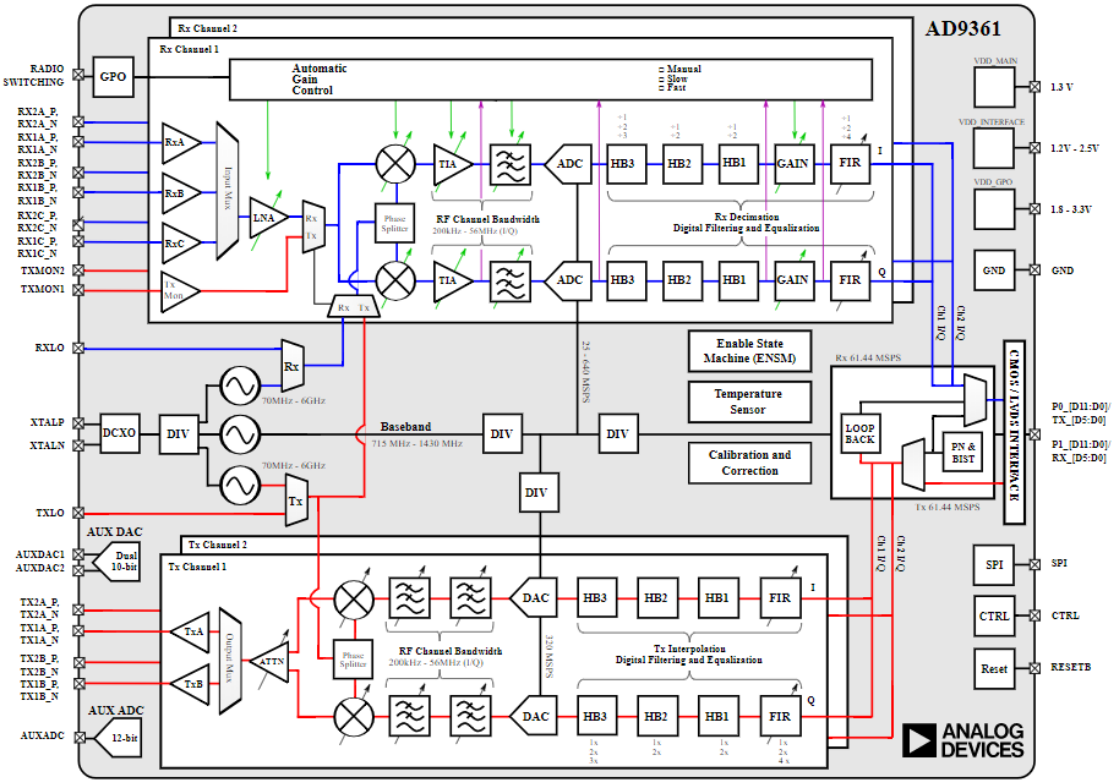


Figure 16. AD9361 IC block diagram. Source: [37].

4. Convert Board

The purpose of the convert board is to upconvert the 2.5125 GHz IF signal output from the SDR to the intended 8.2125 GHz X-band frequency. The NPS payload design course replaced traditional large-scale RF components utilized by Bischoff [27] with drop-in RF components from X-Microwave (X-MW) [25]. X-MW provides standardized drop-in parts, known X-MWblocks, measuring 0.535 inches by 0.535 inches and featuring pre-drilled holes in each corner as seen in Figure 17 [38]. These same parts are also offered in a connectorized housing option, as shown in Figure 18.

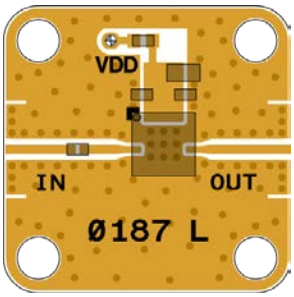


Figure 17. XM-A9W8-0404C-01 X-MWblock. Source [39].

Figure 18. XM-A9W8-0404C-01 connectorized housing. Source [39].

X-MWblocks are attached through a solderless high frequency interconnect afforded by anchors and a ground-signal-ground Jumper (gsgJumper) [40]. Although RF blocks can be connected through traditional soldering or wire bonding, the gsgJumper and anchor combination is the most convenient option for rapid prototyping [40]. Figure 19 portrays the X-MW solderless interconnect feature. The gsgJumper is a flex circuit with three ribbon strips that have gold plated diamond particles allowing for a robust RF connection [41]. Two anchor components and one gsgJumper are required to make one solderless interconnect between XMblocks. Although the solderless interconnect is superb for prototyping purposes, the fragile nature of the connecting hardware prevents it from being utilized in flight-like applications.

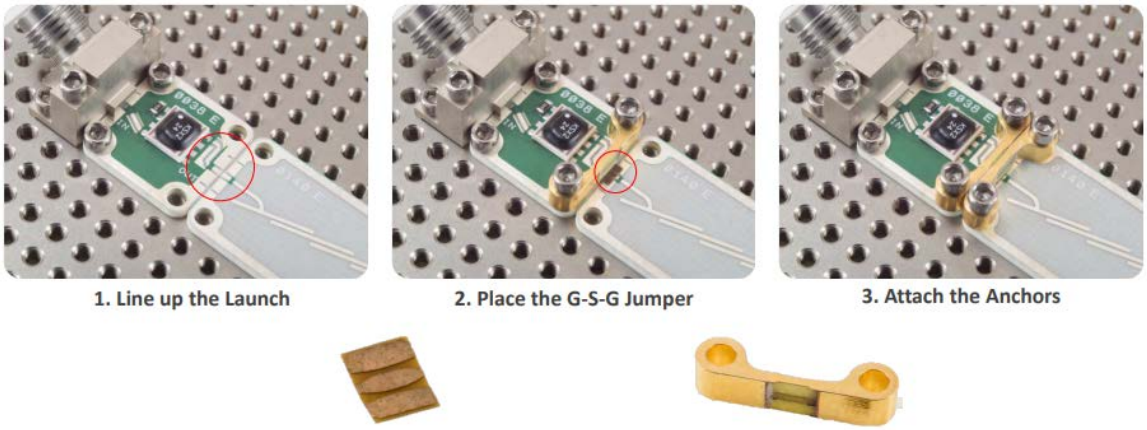


Figure 19. Launch-to-launch solderless interconnect. Source [38].

The design for the convert board was developed on the X-MW online system layout and cascade tool, and simulated using the Keysight PathWave RF Synthesis (Genesys). This web-based design approach allows the user to seamlessly add drop-in RF blocks, drop-on anchors, walls and lids, and automatically adds gsgJumpers and bias controllers to match associated X-MWblocks. The initial layout for the convert board can be seen in the Figure 20.

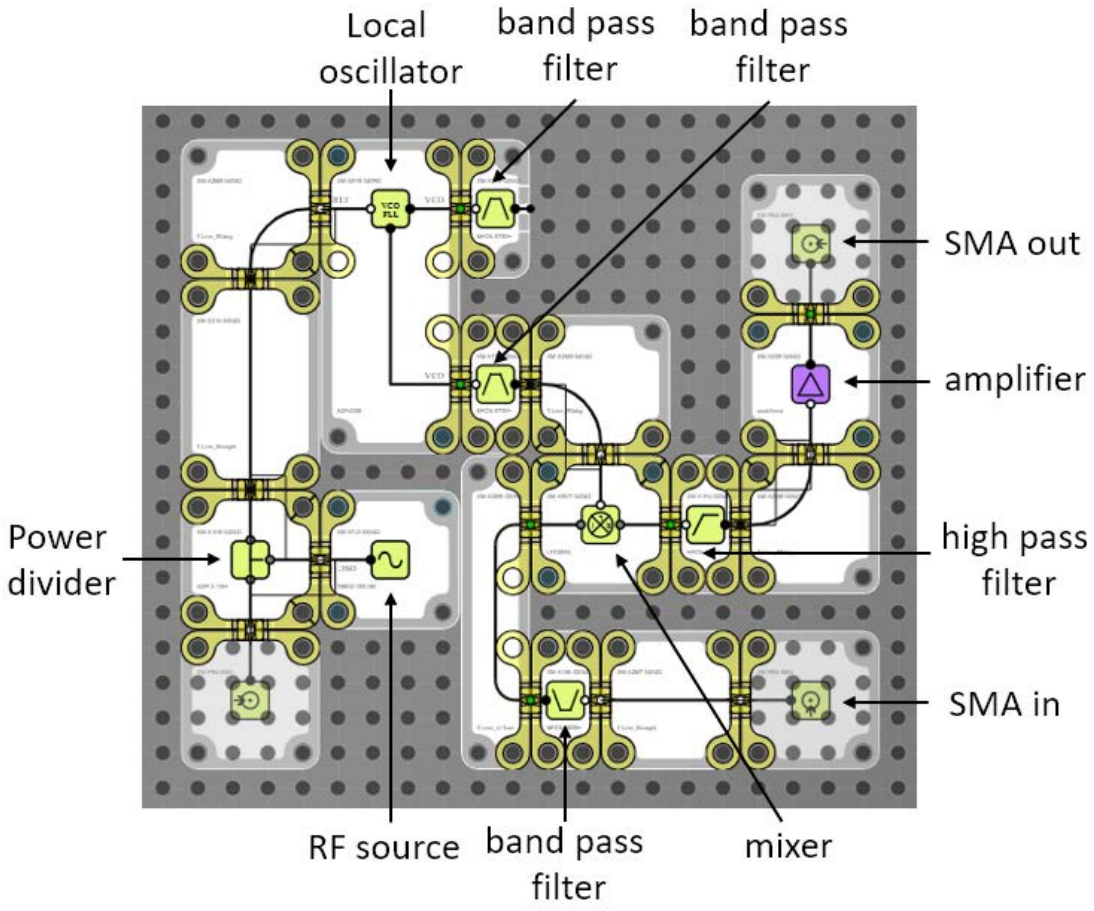


Figure 20. X-MW convert board top design on layout and cascade tool.

Voltage regulation, sequencing, bias and control are provided on the bottom side of the prototyping plate from a matched bias and control block identified in the X-MW layout and cascade tool [40]. Signals are passed from these controllers to the RF board through pre-drilled holes in the prototyping plate, and attached via spring-pins or soldered, 30 gauge, solid core wire with built in strain relief [40]. The layout for the bottom side of the convert board is shown in Figure 21.

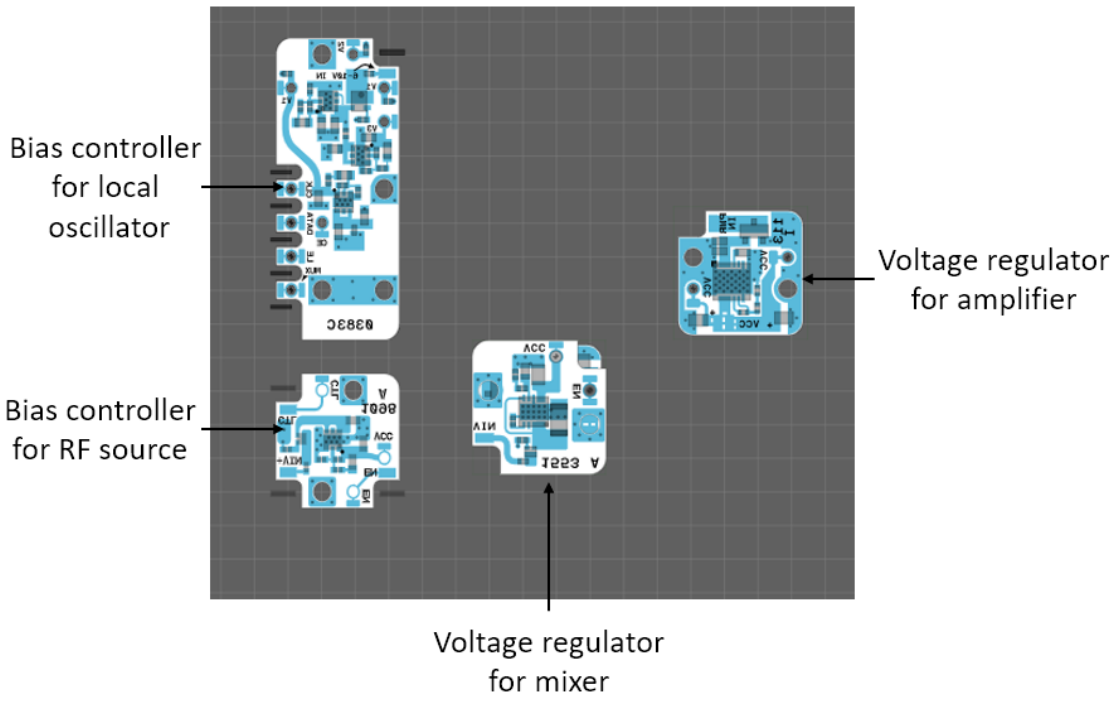


Figure 21. X-MW convert board bottom design on layout and cascade tool.

Once the design was finalized, prototyping began on the X-MW prototyping plate, which is a board that allows for simple construction of X-MW RF components through the use of 1–72 inch threaded holes in a square pattern, spaced 0.135 inches by 0.135 inches apart [42]. The prototyping plate is shown in Figure 22.

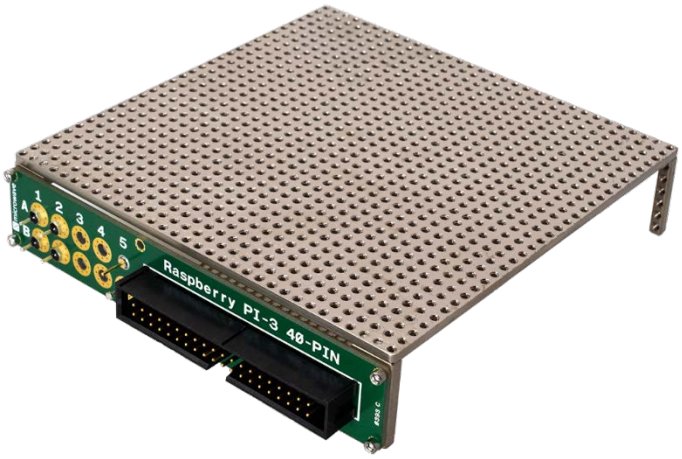


Figure 22. X-MW prototyping plate. Source: [43].

The initial convert board prototype was constructed using the solderless interconnect method. Figure 23 shows the top side of the convert board with X-MWblocks, while Figure 24 shows the bottom side with voltage regulators, bias controllers, and wiring for signaling and power. The initial convert board measured approximately 6.53 by 7.20 cm and weighed approximately 0.02 kg, while the mechanical enclosure used to provide RF isolation of the board measured 8.9 by 8.5 by 0.85 cm and weighed 0.19 kg [25].

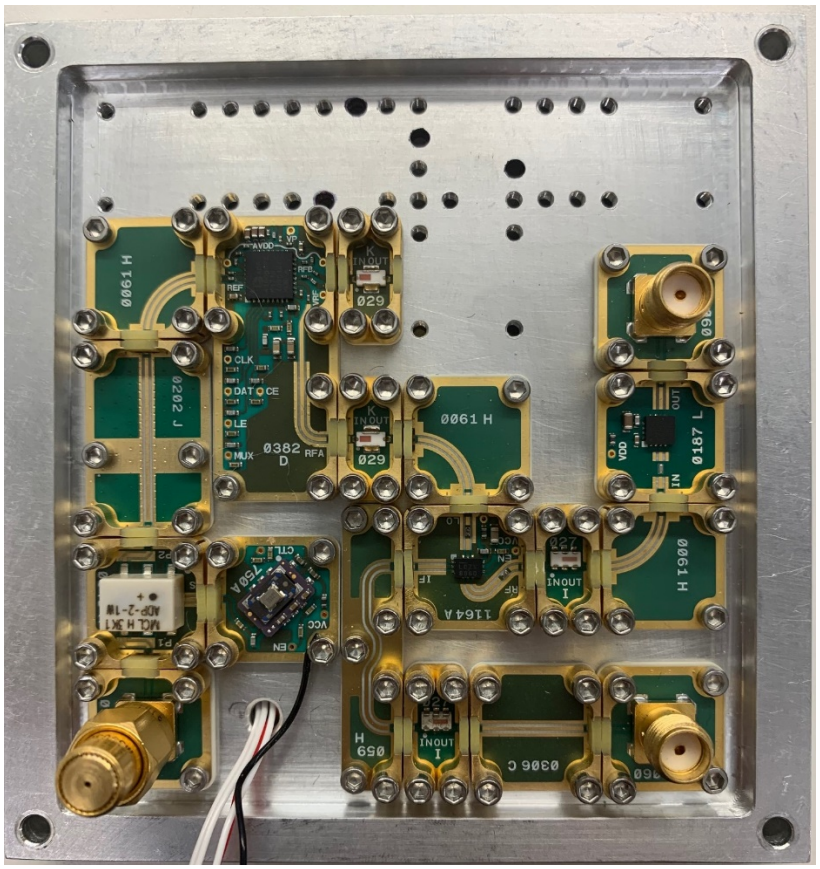


Figure 23. Constructed X-MW convert board top.

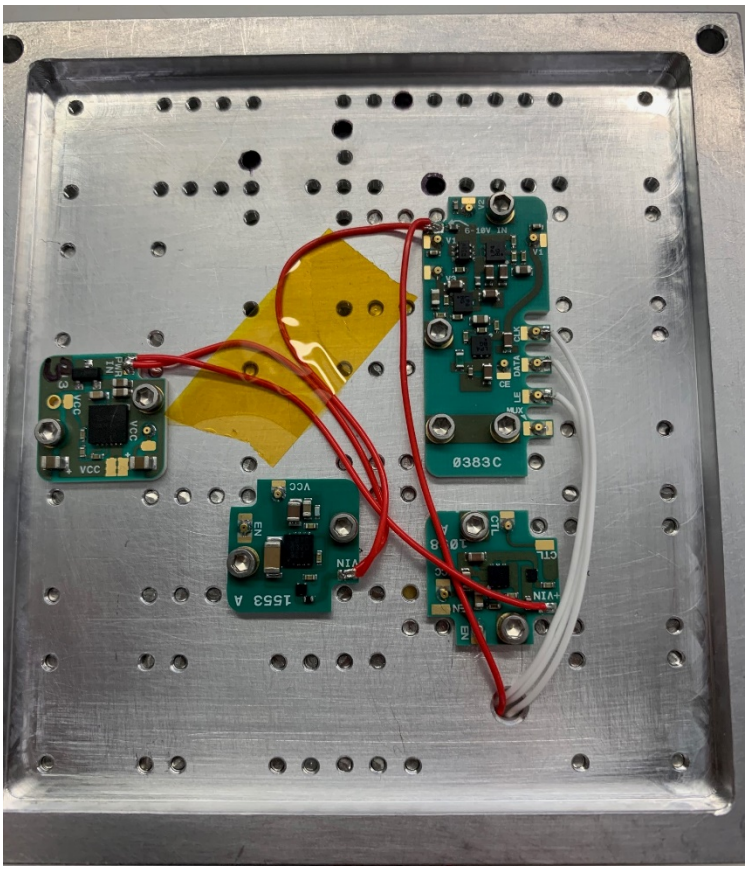


Figure 24. Constructed X-MW convert board bottom.

To upconvert the IF signal, the local oscillator (LO) on the convert board provides an RF signal at 6.8 GHz, to be mixed with the IF signal at 2.5125 GHz from the SDR. The LO must first be programmed to provide the expected RF signal. This task is accomplished by sending data, clock, and LE signals to the ADF4356 wideband synthesizer with integrated voltage-controlled oscillator (VCO). Register values must be established to program the wideband synthesizer and achieve the desired up-conversion. The ADF4356 control software was used to determine the register values with reference to the ADF4356 data sheet [44]. The ADF4356 evaluation board was used for prototyping during this process because it features the same wideband synthesizer found on the X-MW LO and is much easier to configure rapidly. Each time power is cycled, the wideband synthesizer must rewrite register values. Figure 25 shows the ADF4356 evaluation board being

programmed while Figure 26 shows the control software used to establish register values. Table 6 lists these register values.

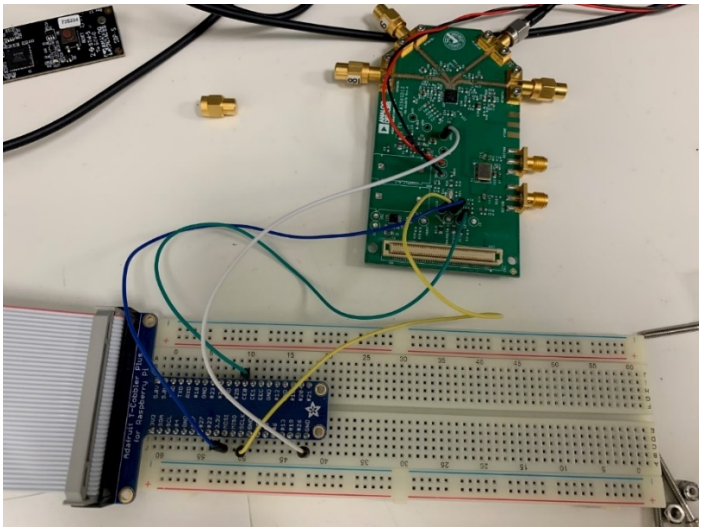


Figure 25. ADF4356 evaluation board programming.

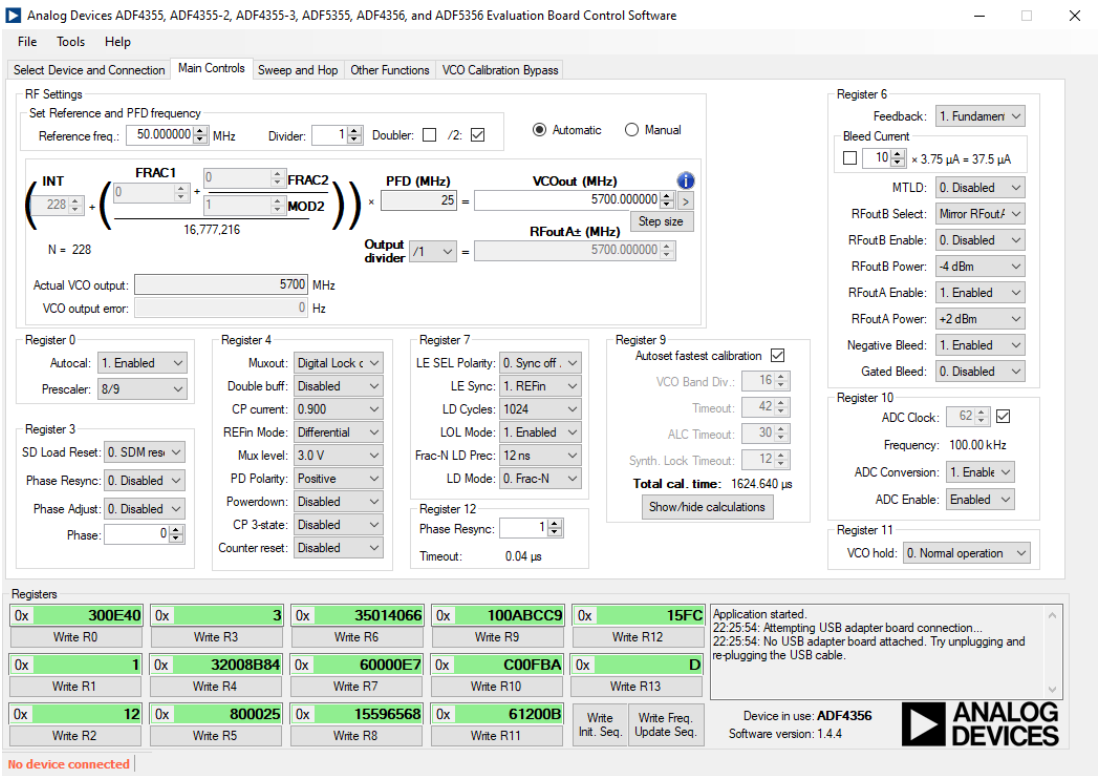


Figure 26. Screenshot of ADF4356 control software.

Table 6. ADF4356 register values.

Register Number	Hex Value
0	300E40
1	1
2	12
3	3
4	32008B84
5	800025
6	35014066
7	60000E7
8	15596568
9	0B0ABCC9
10	C00FBA
11	61200B
12	15FC
13	D

Once the register values were established utilizing the ADF4356 evaluation board, they were sent to X-MW LO on the convert board through the X-MWcontroller touch interface, which offers one-touch configuration for all X-MWblocks shown in Figure 27 [45].

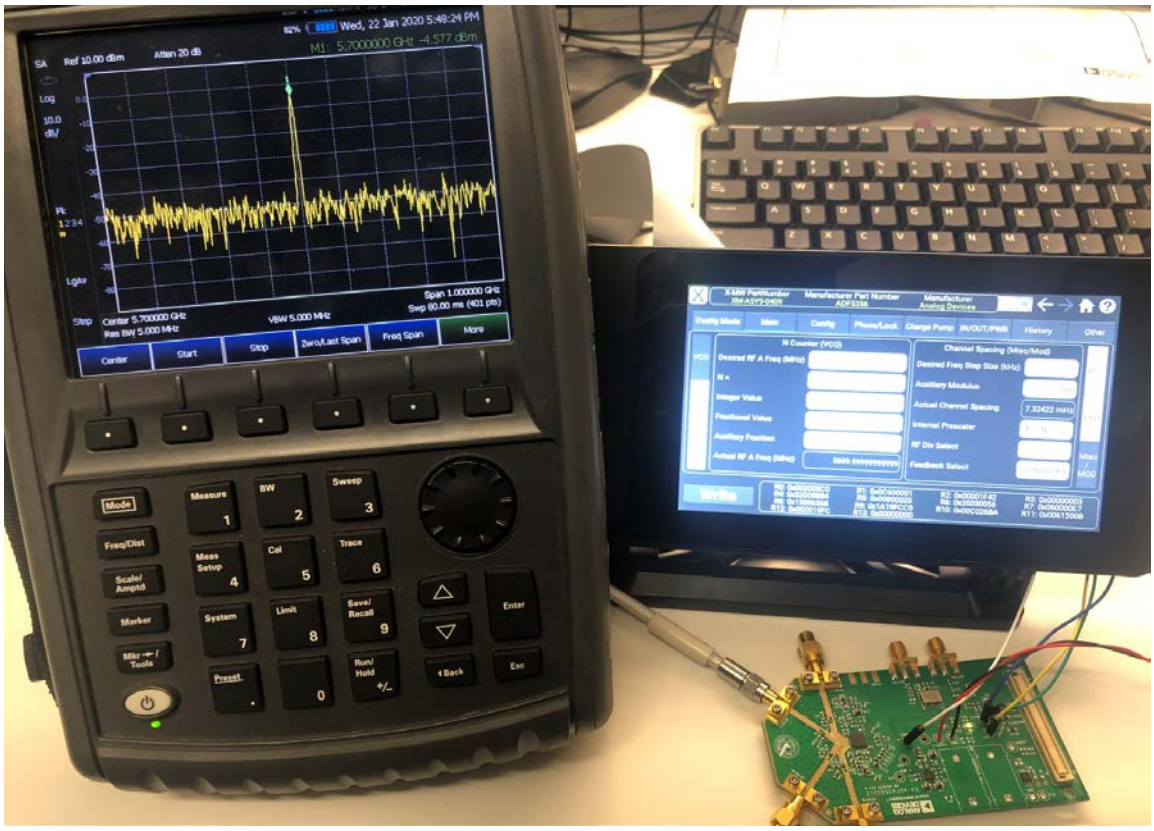


Figure 27. X-MWcontroller programming ADF4356.

5. High-Power Amplifier

The Analog Devices HMC1121 was selected as the baseline high-power amplifier (HPA) for the X-band SDR payload to amplify the power output of the signal for downlink. While this HPA was accounted for in both link budget analyses, it was not included in testing for this research as it is not required for functional testing of the RF design. The HMC1121 HPA is a 4 W power amplifier with an operating range of 5.5 to 8.5 GHz, 28 dB of gain, 44 dBm output, and 36.5 dBm saturated output at 30% power added efficiency (PAE) [46]. The HMC1121 evaluation board is shown in Figure 28.

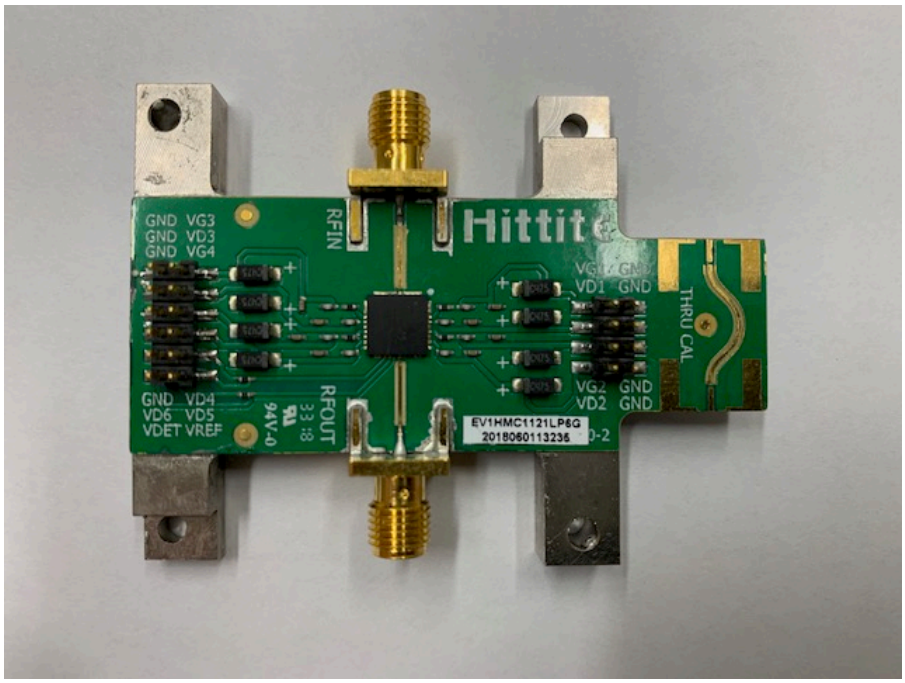


Figure 28. HMC1121 HPA evaluation board.

B. FINAL HARDWARE DESIGN

Although the initial development board, SDR, and convert board met the RF functional requirements for the X-band SDR payload, the form factor of these components prevented their implementation into the EDU. The initial hardware design was over the 0.5U form factor requirement. The ZedBoard/ADFMComms3-EBZ SDR combination measured approximately 22.33 by 13.5 cm and with a mass of 0.2315 kg, composed a significant portion of the overall mass budget of 1 kg. Similarly, the initial convert board enclosure measured 8.9 by 8.5 by 0.84 cm and weighed 0.21 kg with RF components installed. As such, the large initial size and weight dictated the use of smaller form factor hardware, with equivalent or better functionality.

1. SoM

The Analog Devices ADRV9361-Z7035 system on a module (SoM) was selected as the SDR platform for this research. This SoM incorporates the same AD9361 integrated RF agile transceiver featured in the ADFMComms3-EBZ SDR as well as the Zynq-7000

AP SoC family of processing power featured in the ZedBoard at a fraction of the form-factor (100 mm by 62 mm) [47] with a mass of 0.0555 kg, as shown in Figure 29.

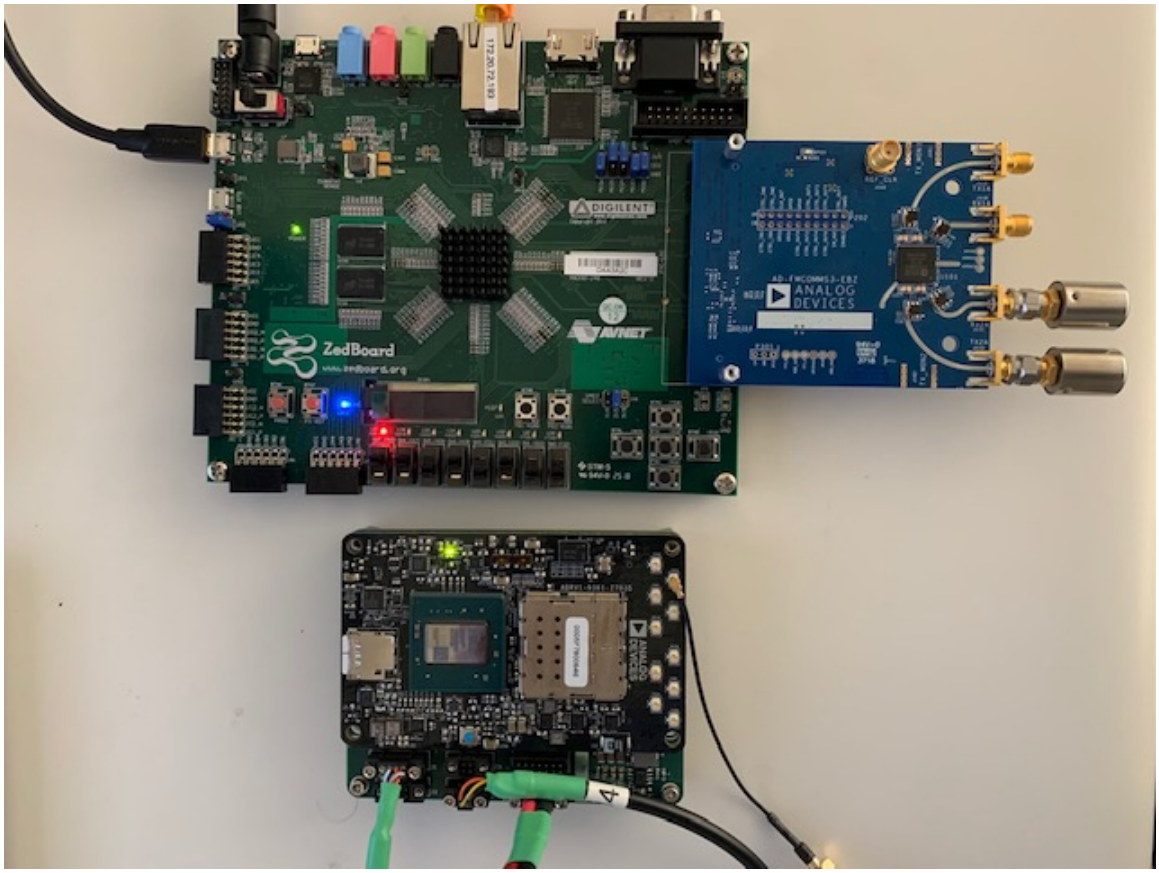


Figure 29. ZedBoard/ADFMComms3-EBZ SDR (top), compared with SoM (bottom).

Additionally, this SoM provides upgraded processing power with a Z7035 variant SoC at a cost-effective price point of $1,300. The SoM features 1 GB DDR3L SDRAM, 256 Mb QSPI flash, microSD card interface, and four transmit and four receive ports through U.FL connectors [47]. The reduced size and weight of the SoM, while maintaining the key features of the ZedBoard, make it an ideal COTS choice for SDR implementation in CubeSats. Figure 30 shows a size reference of the SoM, while Figure 31 shows the SoM hardware block diagram.

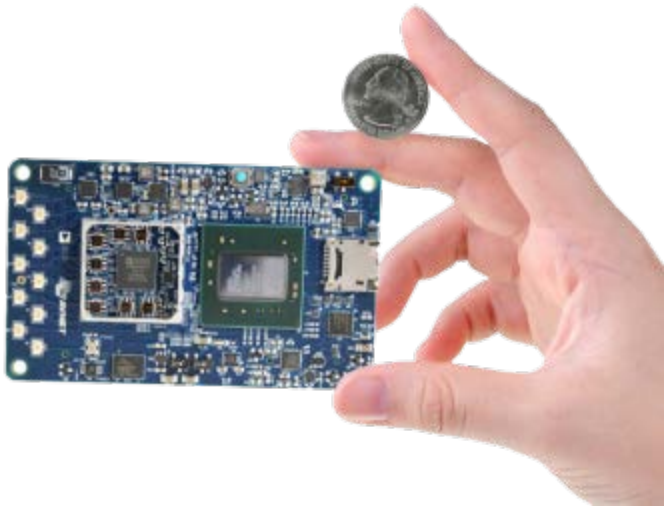


Figure 30. ADRV9361-Z7035 SoM. Source: [48].

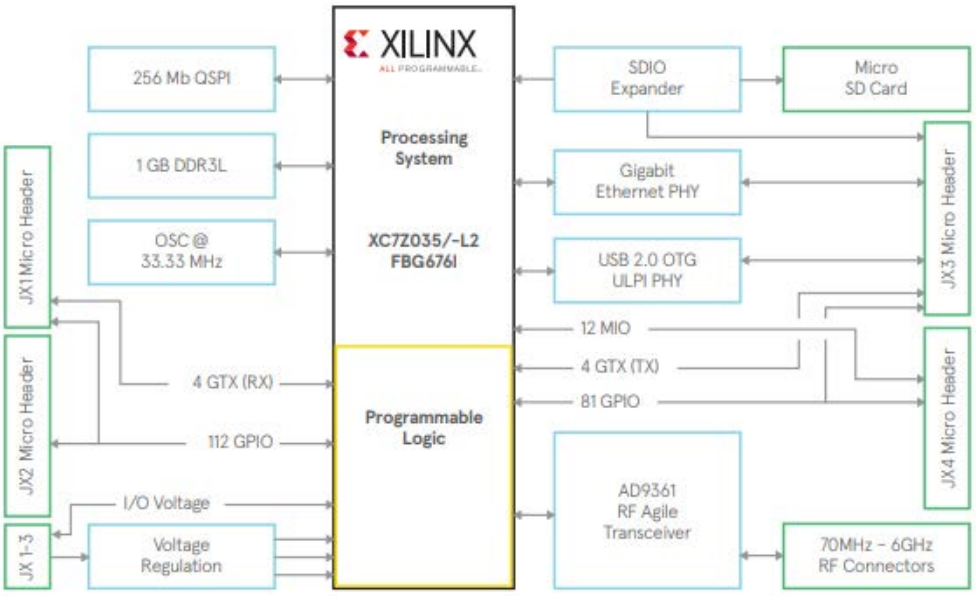


Figure 31. SoM block diagram. Source: [47].

a. SoC

While both the ZedBoard and SoM take advantage of the Zynq-7000 AP SoC family, the SoM incorporates the mid-range Z-7035 variant of the Zynq-7000 SoC. This SoC provides more processing power at speeds up to 1 GHz, 275,000 PL cells, 900 DSP slices, and 17.6 Mb of RAM [33]. The increased processing power, RAM, PL cells, and

DSP slices on this SoC allows for more processes to be run simultaneously and at a faster rate than its Z-7020 counterpart on the ZedBoard. Ultimately, this allows the Z-7035 to better execute a RTOS in a standalone configuration and deliver higher data rates for SDR applications than the initial hardware design.

b. SDR

The SoM features the same AD9361 integrated RF agile transceiver as the ADFMComms3-EBZ SDR with a tuning range of 70 MHz to 6 GHz and supported bandwidth of 200 kHz to 56 MHz. Unlike the initial hardware design, the SoM incorporates the AD9361 directly onto its board without the need for an FMC adapter or the ADFMComms3-EBZ evaluation board.

2. Carrier Board

a. ADRV1CRR-BOB Breakout Carrier

The Analog Devices ADRV1CRR-BOB breakout carrier board is a simplified prototype platform for the ADRV9361-Z7035 SoM. The carrier board features include 10/100/1000 MBps ethernet, USB2.0 OTG, USB-UART, JTAG, 162 user I/O pins, four push buttons, four switches, and four LEDs [49]. The carrier board, weighing approximately 0.065 kg, is shown in Figure 32.

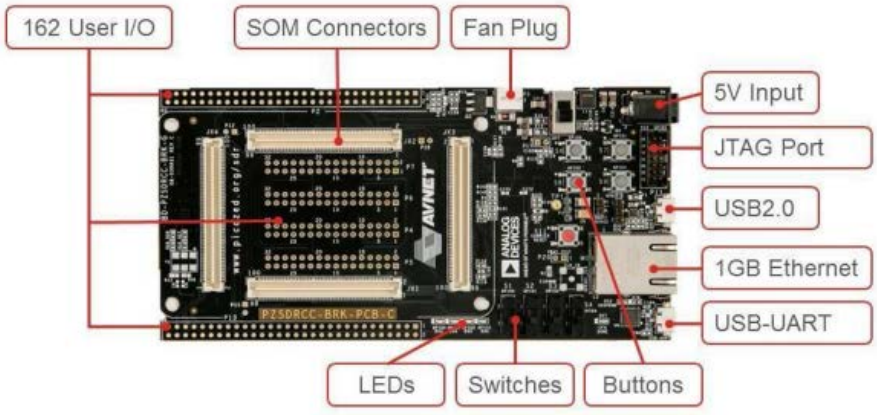


Figure 32. ADRV1CRR-BOB breakout carrier overview. Source: [50].

Although this carrier board was used for prototyping and initial testing of the SoM, its size was determined to be a limiting factor. Mounting the SoM on the carrier board significantly increased the form-factor and created the need for a custom solution. The increased size footprint in relation to the SoM can be seen in Figure 33.

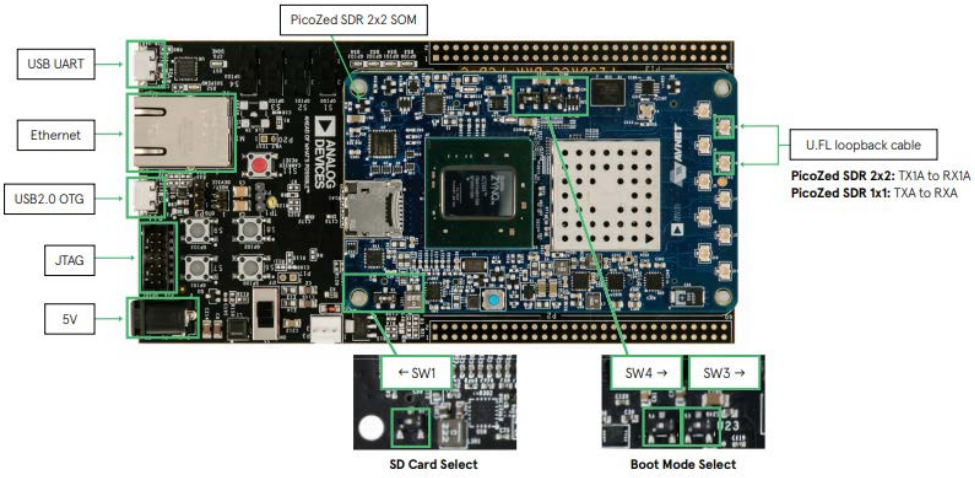


Figure 33. SoM mounted on COTS breakout carrier board. Source: [51].

b. Custom Carrier Board

To maintain a form-factor better suited for implementation into CubeSats, a custom carrier board was developed by the SSAG, which removed unnecessary peripheral I/O ports using the ADRV1CRR-BOB as a reference design. The custom board retained the 10/100/1000 ethernet, USB-UART, USB-JTAG, and power characteristics of its reference design. The printed circuit board (PCB) layout of this carrier is seen in Figure 34, which details connectors JX1 to JX4 used to interface with the SoM. Figure 35 depicts the custom carrier board while Figure 36 shows the SoM installed on the printed custom carrier board. The mass of this custom carrier board was approximately 0.047 kg.

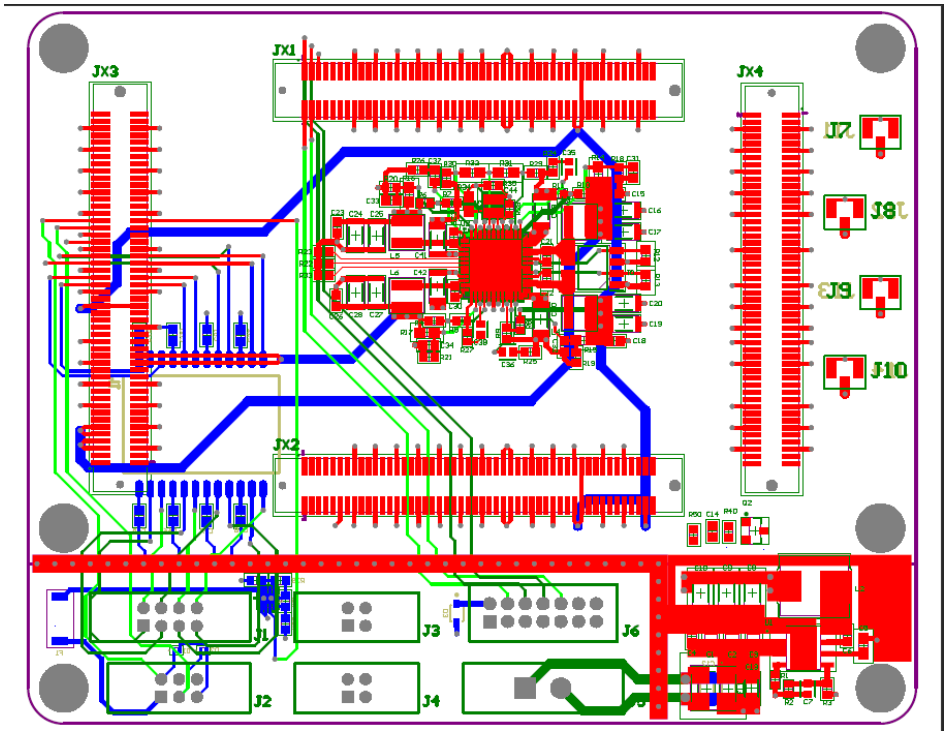


Figure 34. Custom carrier board layout.

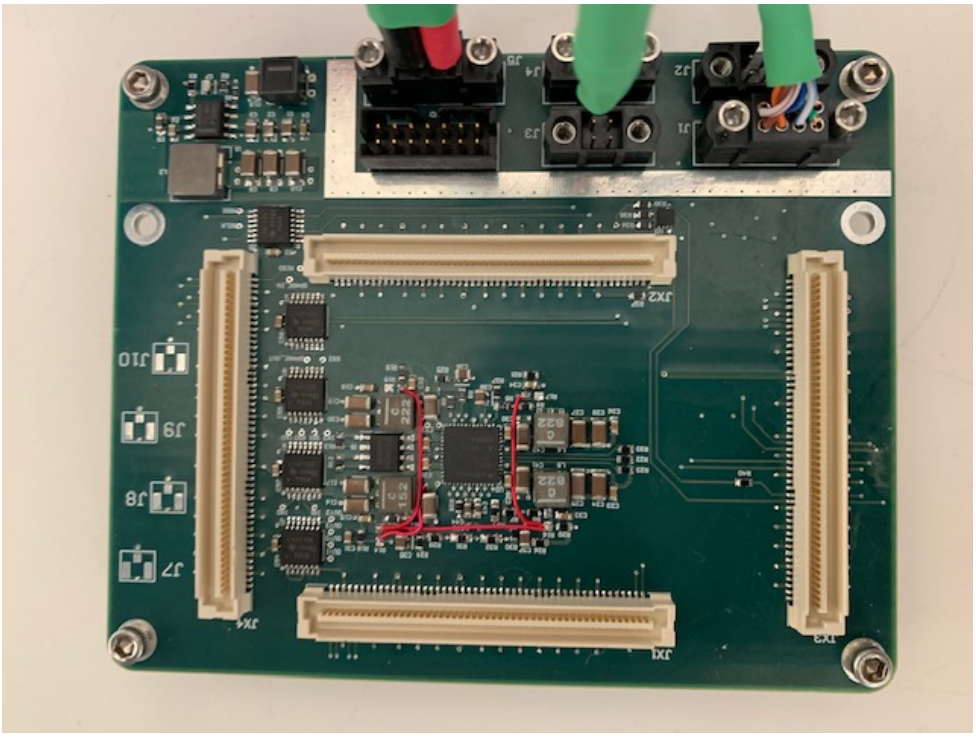


Figure 35. Printed custom carrier board prototype.

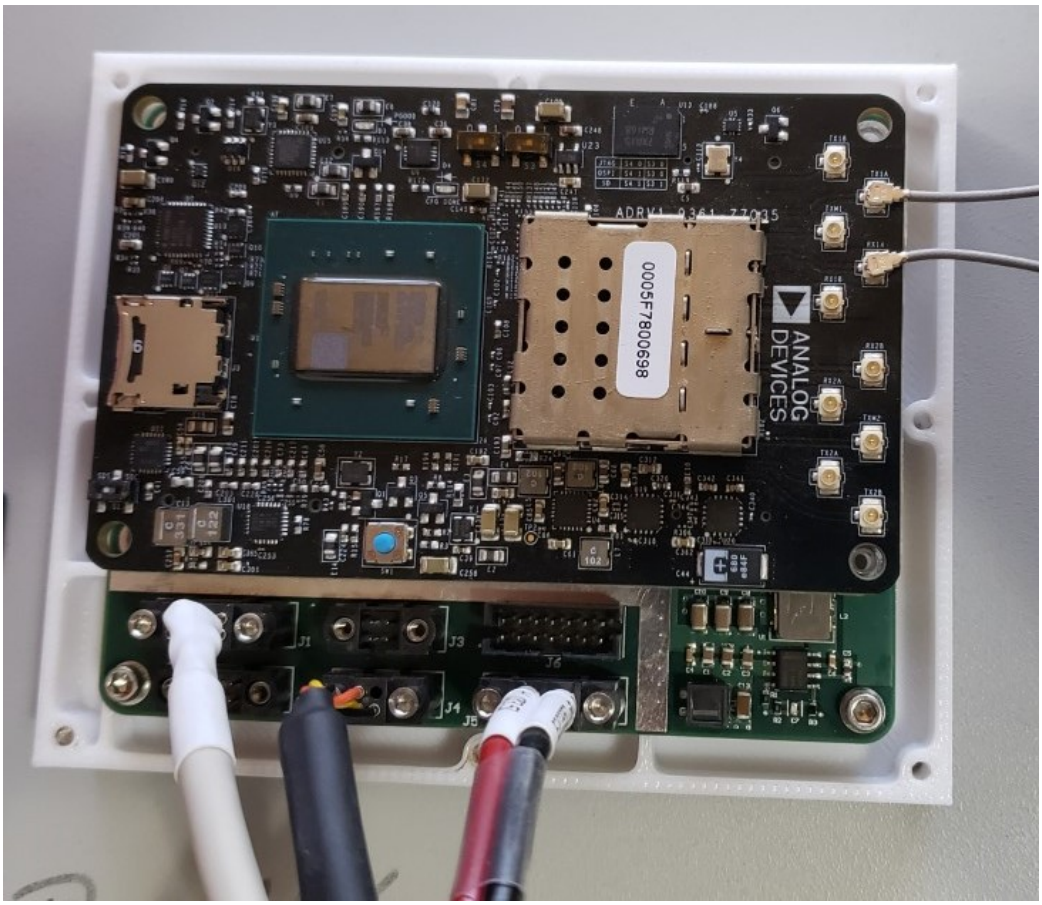


Figure 36. SoM mounted on custom carrier board prototype.

3. Convert Board

Similar to the original carrier board, the initial convert board needed to be redesigned in order to more easily fit into a form-factor compatible with the X-band SDR radio assembly payload mechanical enclosure. Several redundant RF components were removed from the original layout while the design was developed and simulated using the X-MW online tools previously discussed. The XM-A328-0404D amplifier that was used in the initial design was obsolete at the time of this revision, so it was replaced by the XM-A9W8-0404D amplifier per the manufacturer's recommendation. The XM-PB4-SMA XMblock features a threaded female SMA connector, (part number 32K10A-40ML5), developed by Rosenberger [52]. The XM-ANCHOR2 is utilized to create the solderless interconnect between X-MW RF components in conjunction with an XM-GSGJ gsgJumper [53]. The XM-A2R9-0404D is a voltage regulator with an input voltage range of 3.5 to 15

V and an output voltage of 3 V and is used to control the XM-A9W8-0404D amplifier. This regulator features the LP38798SD high-performance low-dropout regulator (LDO) from Texas Instruments [54]. The XM-A9W8-0404D is an amplifier block that has a frequency range of 6000 MHz to 18000 MHz and incorporates the Custom MMIC CMD264P3 low noise amplifier [55]. The XM-A7J3-0404D is an LO block that utilizes the TB602-050.M oscillator from Connor Winfield [56]. Block XM-A7T8-0404D is a bias controller that uses the ADP7142 LDO from Analog Devices and is used to control the XM-A7J3-0404D oscillator [57]. Component XM-A3R9-0409D is a bias controller and also uses the ADP7142 LDO from Analog Devices and is used to control the XM-A5Y8-0409D phase lock loop (PLL). The XM-A5Y8-0409D is a PLL with integrated VCO featuring the ADF4356 6.8 GHz wideband synthesizer from Analog Devices [44]. Block XM-A1F4-0204D is a high pass filter featuring the HFCN-7150+ high pass filter from Mini-Circuits with a frequency range of 7900 to 11000 MHz [58]. XM-A718-0204D is a bandpass filter using the BFCN-5750+ bandpass filter from Mini-Circuits with a frequency range of 5650 to 5850 MHz [59]. The XM-A9V7-0404D is a mixer that incorporates the LTC5553 3 to 20 GHz microwave mixer from Analog Devices [60]. The XM-C6A1-0404D is bias controller featuring the LT3045EDD LDO from Analog Devices and is used to control the XM-A9V7-0404D mixer [61]. Block XM-A166-0204D is another bandpass filter and features the BFCN-2555+ bandpass filter from Mini-Circuits with a frequency range of 2500 to 2610 MHz [62]. The full bill of materials (BOM) for these components in the final design can be seen in Table 7.

Table 7. Convert board BOM. Source: Source: [63].

ID	Part Number	Feature	QTY	Each	Total
1	XM-PB4-SMA	SMA connector	2	$59.95	$119.90
2	XM-ANCHOR2	Anchor	15	$9.95	$149.25
3	XM-GSGJ	gsgJumper	8	$4.00	$32.00
4	XM-A2R9-0404D	Voltage regulator	1	$65.00	$65.00
5	XM-A9W8-0404D	Amplifier	1	$110.00	$110.00
6	XM-A7J3-0404D	Oscillator	1	$110.00	$110.00
7	XM-A7T8-0404D	Bias controller	1	$150.00	$150.00
8	XM-A3R9-0409D	Bias controller	1	$180.00	$180.00
9	XM-A5Y8-0409D	PLL VCO	1	$204.00	$204.00
10	XM-A1F4-0204D	High pass filter	1	$44.00	$44.00
11	XM-A718-0204D	Band pass filter	1	$44.00	$44.00
12	XM-A9V7-0404D	Mixer	1	$199.00	$199.00
13	XM-C6A1-0404D	Bias controller	1	$65.00	$65.00
14	XM-A166-0204D	Band pass filter	1	$44.00	$44.00
	TOTAL (USD)				**$1516.15**

The design for the new convert board is displayed in Figure 37. The ID number from each X-MW part in the BOM corresponds to the number displayed in the X-MW cascade and layout tool.

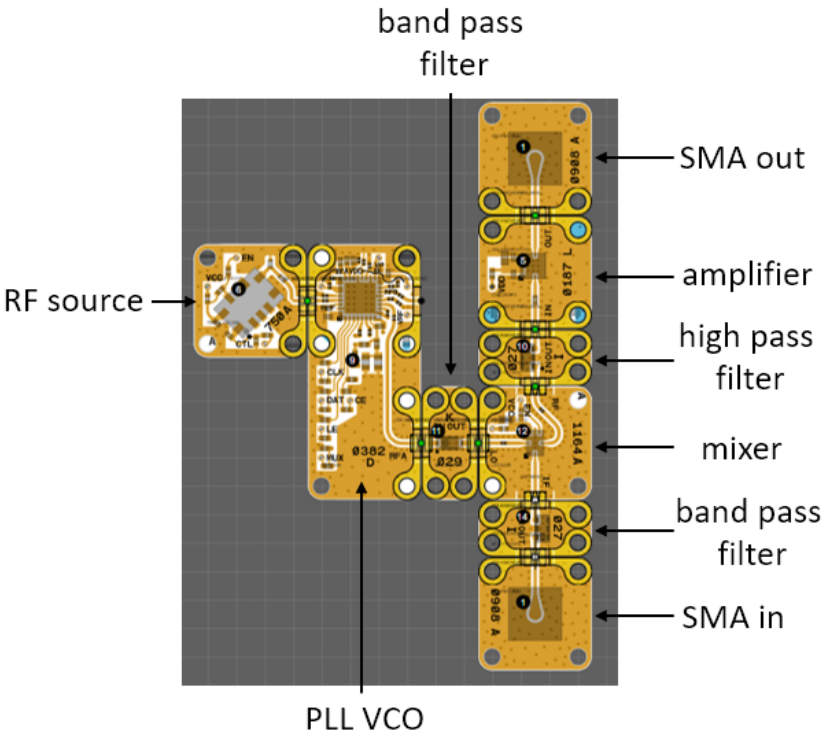


Figure 37. New X-MW convert board top design.

The same voltage regulators and bias controllers were used for both iterations of the convert board, albeit in a new configuration. This new configuration is seen in Figure 38.

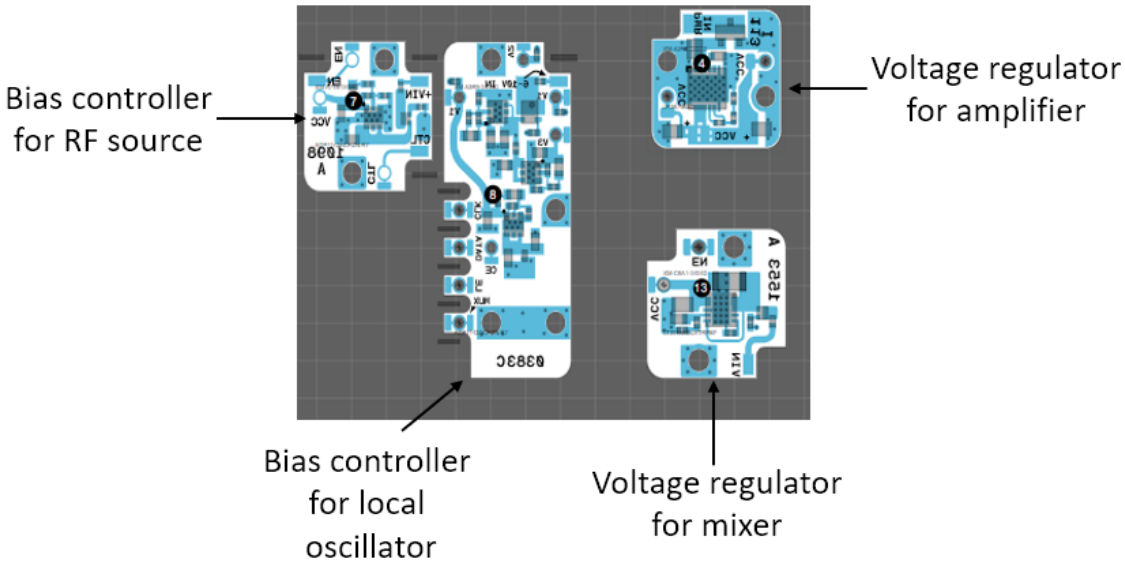


Figure 38. New X-MW convert board bottom design.

After the design was simulated, X-MWblocks drop-in RF components were laid out and connected on the X-MW prototyping plate using the X-MW solderless interconnect system. Register values were confirmed using the ADF4356 evaluation board and programmed to the convert board via the X-MWcontroller touch interface panel. The new convert board construction top side is shown in Figure 39. Figure 40 depicts the wiring from bias controllers and voltage regulators on the bottom side of the convert board.

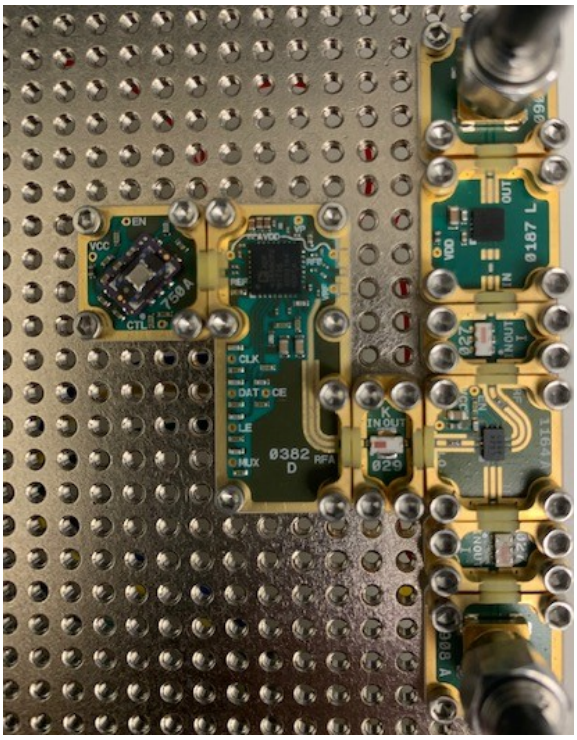


Figure 39. New constructed X-MW convert board, top.

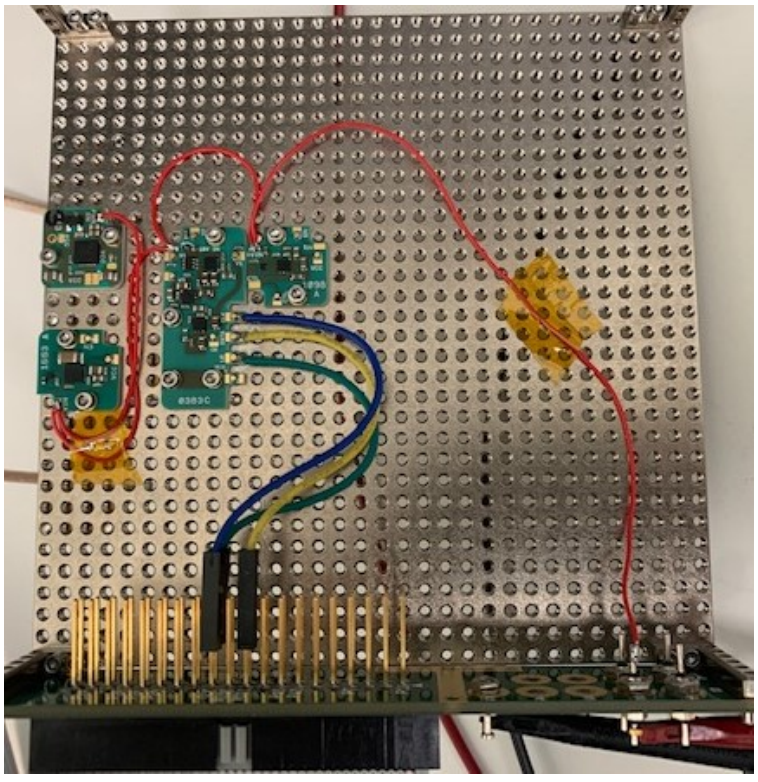


Figure 40. New constructed X-MW convert board, bottom.

4. Mechanical Enclosure

The radio assembly mechanical enclosure was developed by the SSAG to incorporate the custom carrier board, SoM, and convert board into an approximately 0.5U CubeSat form-factor payload, measuring 9.39 by 11.46 by 5.63 cm. This enclosure will provide RF isolation of each board of the X-band SDR payload and accommodate interfaces with the commercially provided 6U bus. Figure 41 shows the assembled mechanical enclosure while Figure 42 shows an expanded view.

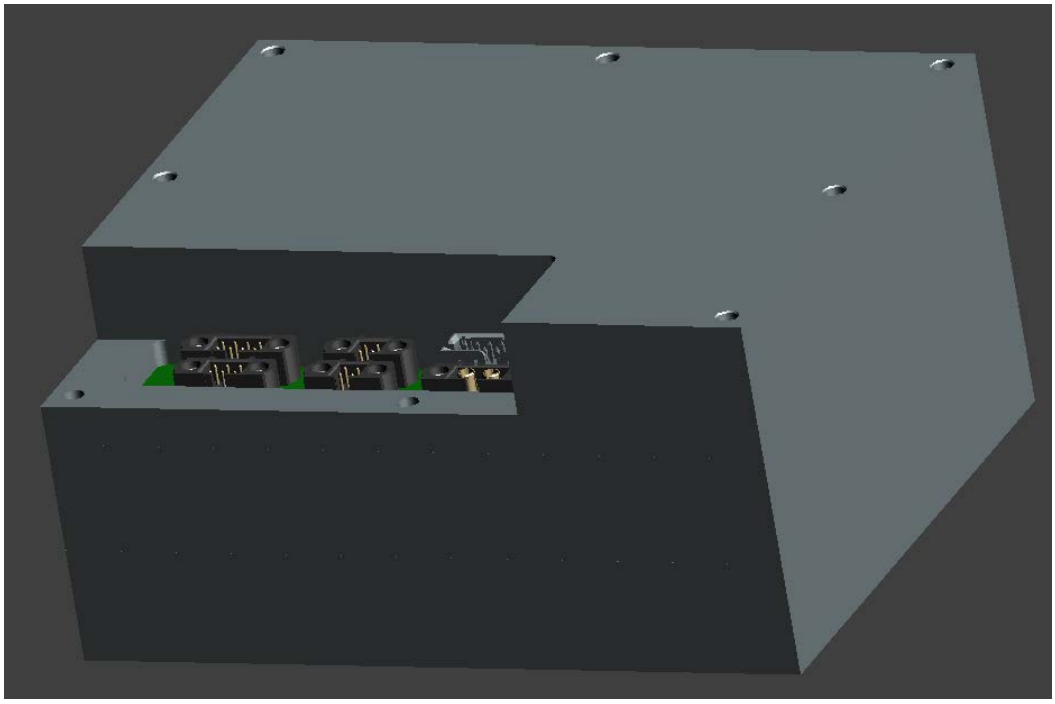


Figure 41. Radio assembly mechanical enclosure.

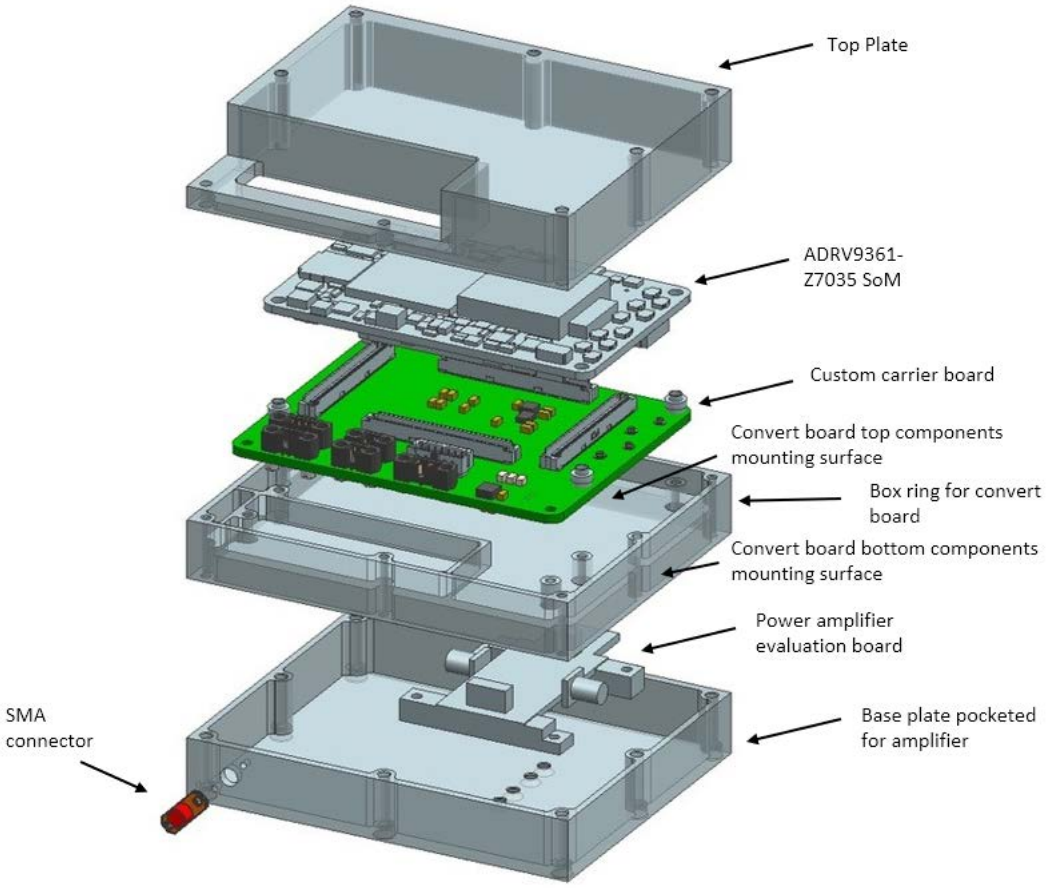


Figure 42. Radio assembly mechanical enclosure expanded view.

IV. SOFTWARE

The requirements for high data rate transmission, digital modulation, and low-cost implementation drove the decision to use readily available software to program and operate the X-band SDR. MATLAB toolboxes enable seamless generation of C and HDL code from Simulink models, which can be implemented directly onto the SoC. An embedded RTOS on the SoC allows users to generate and deploy code in the hardware, to operate the system as a standalone rather that conducting signal processing on a host computer. Additionally, MATLAB Simulink provides reference models for transmit and receive using QPSK modulation. Unlike GNU radio, which is open-source, not clearly documented, and requires intricate knowledge of Python, C ++, and Linux, MATLAB and Simulink are user-friendly, well documented, and do not require background knowledge of other software or programming languages to manipulate efficiently.

A. INITIAL SOFTWARE DESIGN

To develop and integrate the SDR, it is important to first build accurate system models and simulate system behavior. For the initial design iteration, this research focused on embedded software applications on the SoC to create a standalone RTOS. MALTAB Communications System Toolbox, Signal Processing Toolbox, DSP System Toolbox, SimRF, Embedded Coder and hardware description language (HDL) Coder allow users to design, analyze, and test SDR systems. A simplified version of the X-band SDR Simulink model from the NPS payload design course (Appendix C) was developed for this initial system as shown in Figure 43.

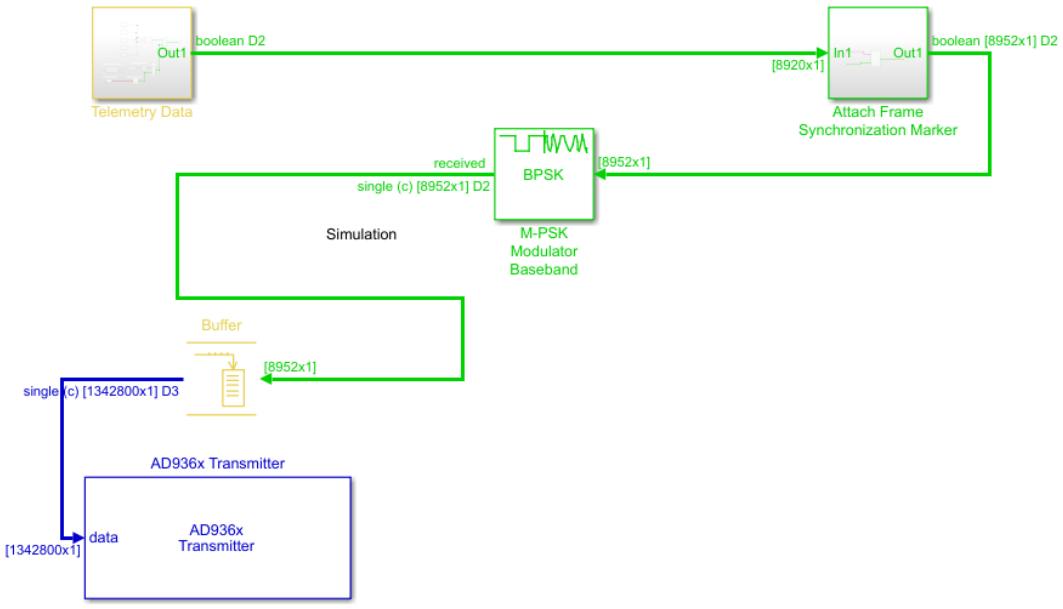


Figure 43. Simplified Simulink SDR model.

A PN11 sequence bit stream was generated and buffered to simulate telemetry data. A frame synchronization marker was then attached before the stream was modulated using BPSK. Modulated data is then buffered again and sent to the AD936x Transmitter, which incorporates the SimRF AD9361 model. The SimRF model of the AD9361 integrated RF agile transceiver replicates the exact functionality of the AD9361, as shown in Figure 44. The workflow used for this research design process is demonstrated in Figure 45.

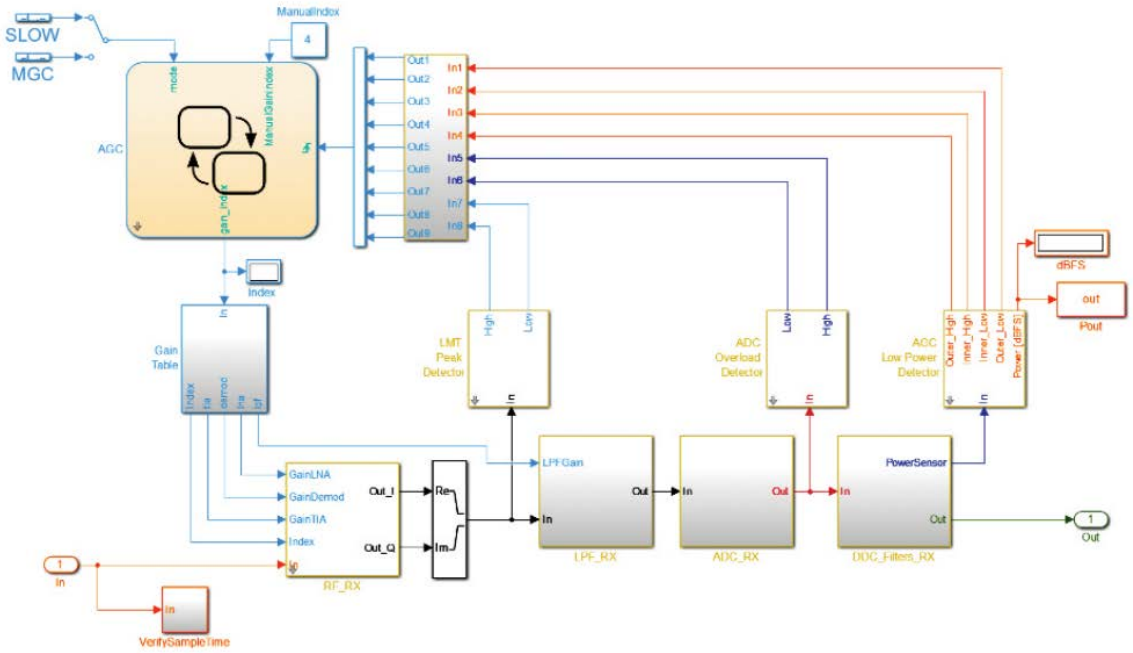


Figure 44. MATLAB SimRF AD9361 model. Source: [34].

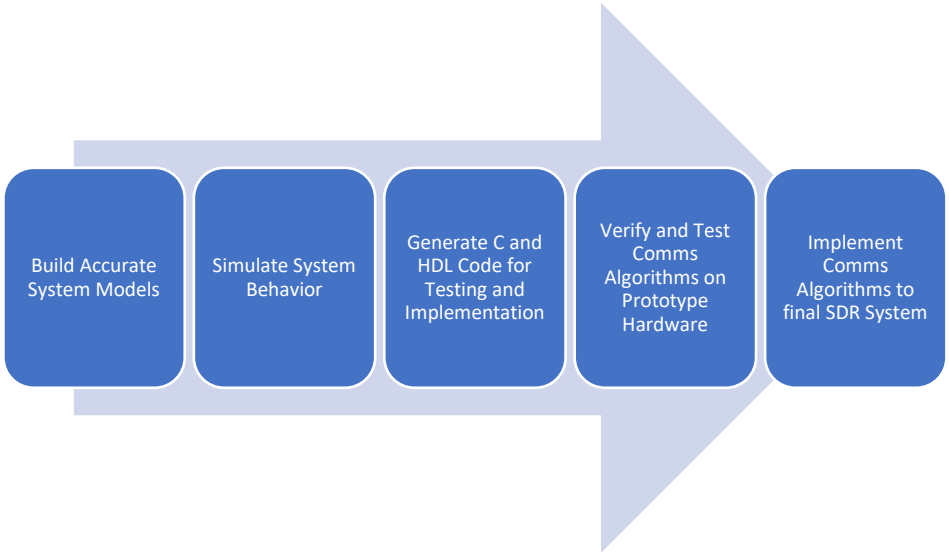


Figure 45. Communications design workflow.

Following full verification of the simulated environment, the Embedded Coder and HDL Coder support packages from MATLAB allow the user to generate C code and Verilog HDL to deploy to hardware for prototyping and testing. Automatic code generation from these MATLAB tools minimizes manual user coding errors to ensure implementation

matches the actual SDR model. Once verified, HDL code can be generated from the Simulink model using the HDL coder tool and C code can be generated from the Embedded Coder tool to prepare for platform deployment. Analog Devices provides a Vivado HDL reference design and the Workflow Advisor implementation tool, allowing the user to transfer HDL into the Xilinx Vivado Design Suite. Vivado will synthesize and analyze HDL designs and then generate and package the bitstream for deployment to the SDR hardware. The bitstream is a file composed of a sequence of bits, which contains the programming information for the SoC. Once Vivado has exported the hardware design to the hardware specification file, or Xilinx Shell Archive (XSA), the Vitis integrated design environment (IDE) is used to export the XSA file to the workspace to generate a standalone domain and first stage boot loader (FSBL) application project. The workflow from this process is depicted in Figure 46.

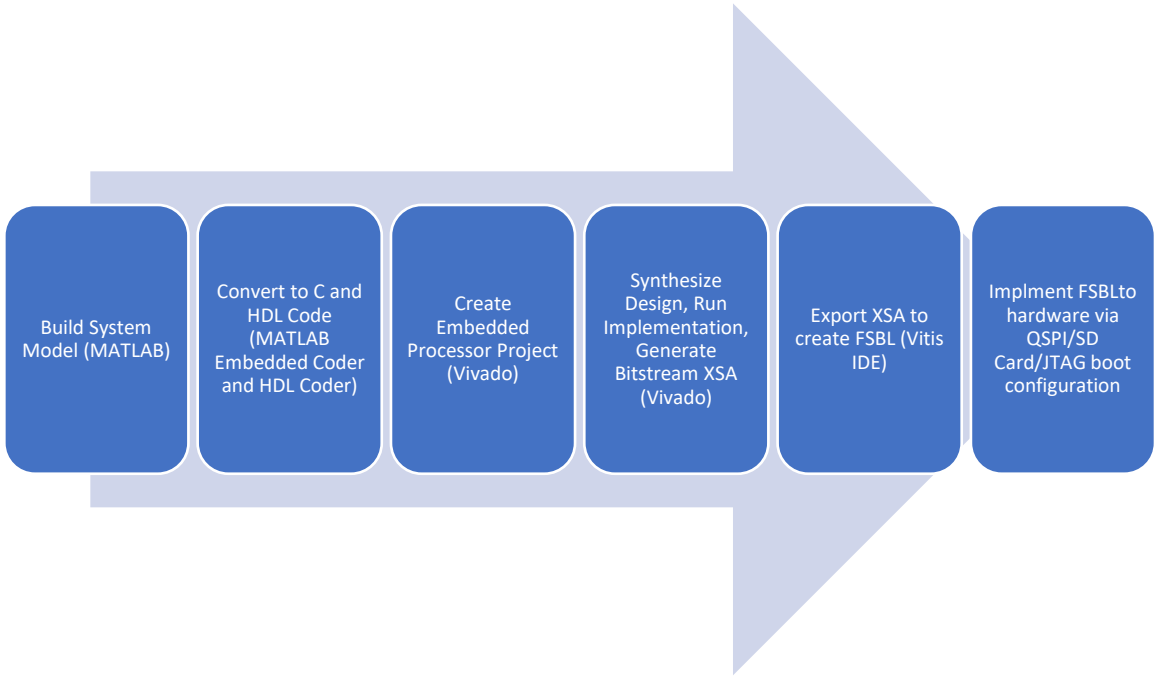


Figure 46. Embedded processor project design and implementation.

Since the payload will be on-orbit, changes in SDR configuration need to be implemented by either flashing the QSPI or rewriting the SDR card. If the QSPI is flashed,

the SoC will first program itself with the contents found in the QSPI flash memory. If the QSPI is not flashed, the processor will then boot from the SD card. The boot method is first selected by the user, then the boot ROM will search the boot media for a BOOT.BIN file and attempt to execute the FSBL. The FSBL will then configure the PL if a bitstream is found and hand over system execution to the application executable [64].

Research conducted during this project was unable to successfully implement a bitstream onto the SoC due to errors encountered working in Vivado Design Suite. At the time of this research, the author was utilizing a student product license, limiting the functionality of Vivado and disabling key features required for exporting an embedded processor project. The author elected to instead to focus on MATLAB provided Simulink models for QPSK transmit and receive, to prepare for functional testing of the hardware/software.

B. FINAL SOFTWARE DESIGN

This research implemented the MATLAB Simulink QPSK transmitter and receiver models to operate the SDR and to perform follow-on functional testing. The models are found within the documentation for the Communications Toolbox Support Package for Xilinx Zynq-Based Radio version 19.2.2. The transmit model continuously sends an indexed “Hello World” message, that is QPSK-modulated, onto a carrier with a predetermined center frequency. In turn, the receiver model demodulates the transmitted message and displays results of the message within the diagnostics window. Before executing the transmit and receive QPSK Simulink models, it is important to first calibrate the frequency offset of the system using the matched pair of models.

1. Frequency Calibration Transmit Model

This calibration transmitter sends a 10 kHz tone at a default center frequency of 2.4 GHz with a baseband sample rate of 520.841 kHz and a gain of -10 dB. For proper calibration, it is important that the transmit calibration model be started before the receive calibration model, and that the baseband sample rate is identical for both transmit and receive [65]. Figure 47 shows the frequency offset calibration transmit model and associated AD9361 block parameters.

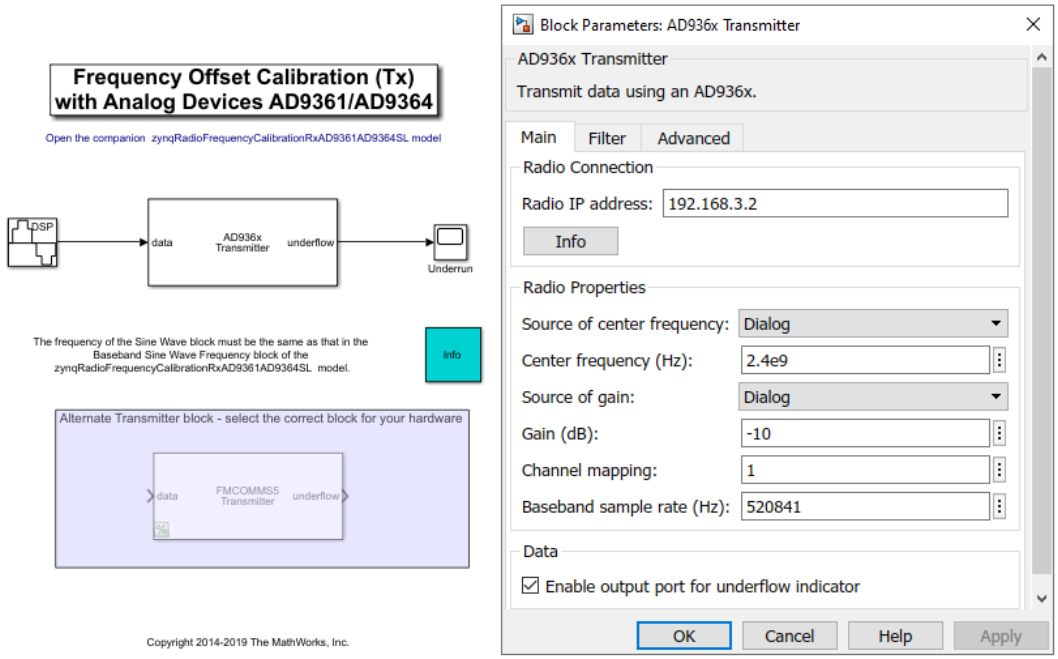


Figure 47. Simulink QPSK frequency offset calibration transmitter.

2. Frequency Calibration Receive Model

The receiver model detects the transmitted tone using a fast Fourier transform (FFT) method. The detected offset from the transmitted 10 kHz tone and the received tone is then calculated and shown in the frequency offset display block of the receiver model. This offset is added back to the center frequency parameter of the receive model, so that frequency offset is driven toward 0 Hz, although there is always noise present in the system and so residual offset will be present. The frequency offset calibration receiver model and associated AD9361 block parameters are shown in Figure 48.

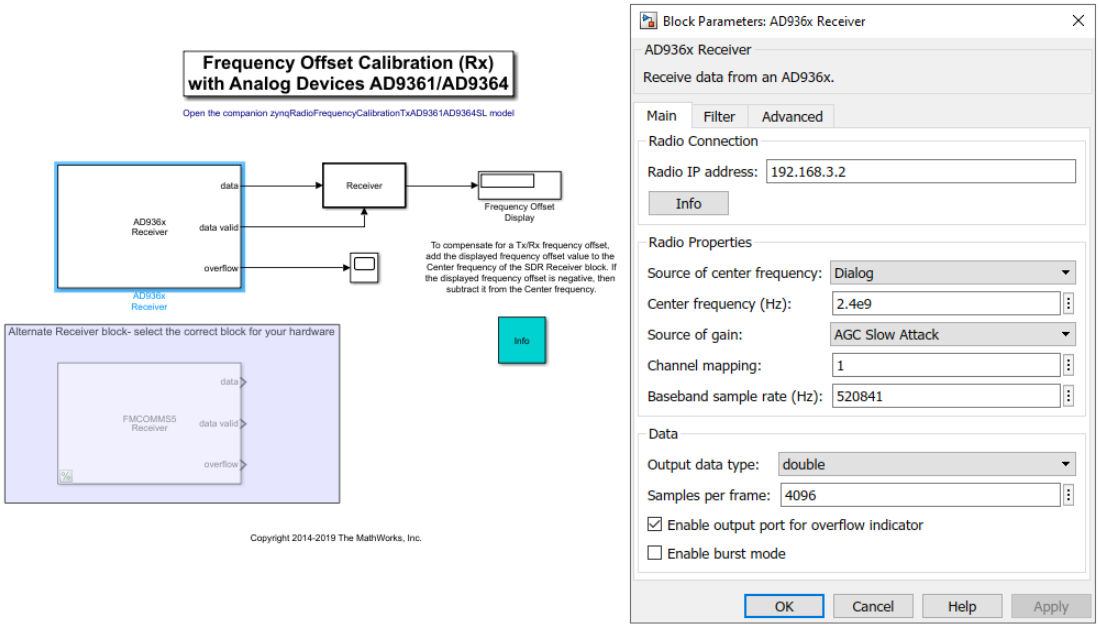


Figure 48. Simulink QPSK frequency offset calibration receiver.

When the receiver model is run, the included MATLAB spectrum analyzer shows the maximum received signal power, as shown in Figure 49.

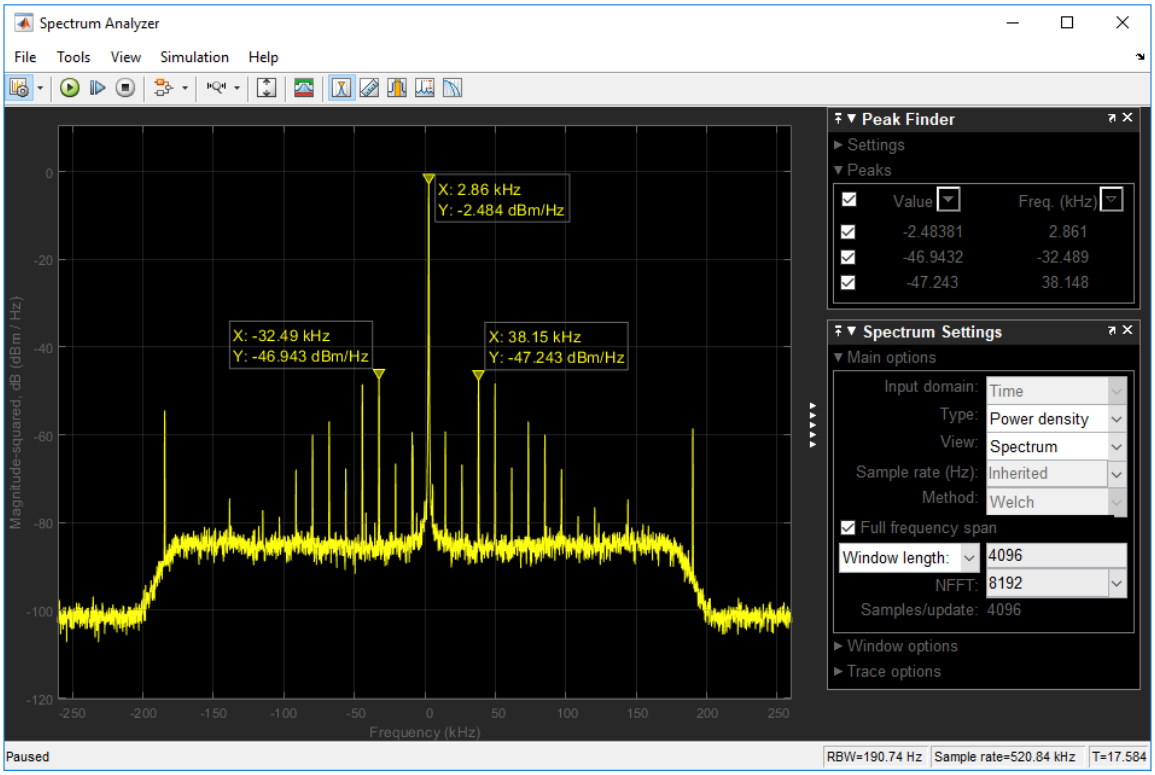


Figure 49. Spectrum analyzer of QPSK frequency offset calibration receiver. Source: [65].

In this example, the maximum received signal power is 2.86 kHz, which is 7.14 kHz less than the transmitted 10 kHz tonal. Adding the 7.14 kHz offset to the center frequency parameter of the receive model will drive the frequency offset toward 0 Hz.

3. QPSK Transmit Model

Following successful frequency offset calibration, the QPSK transmitter using Analog Devices AD9361 is run with a default center frequency of 2.4 GHz and baseband sample rate of 520.841 kHz. This Simulink model transmits a QPSK signal using the SDR hardware through bit generation, baseband modulation, pulse shaping and up-sampling, and sending baseband data to the SDR hardware. Figure 50 shows the QPSK transmitter model.

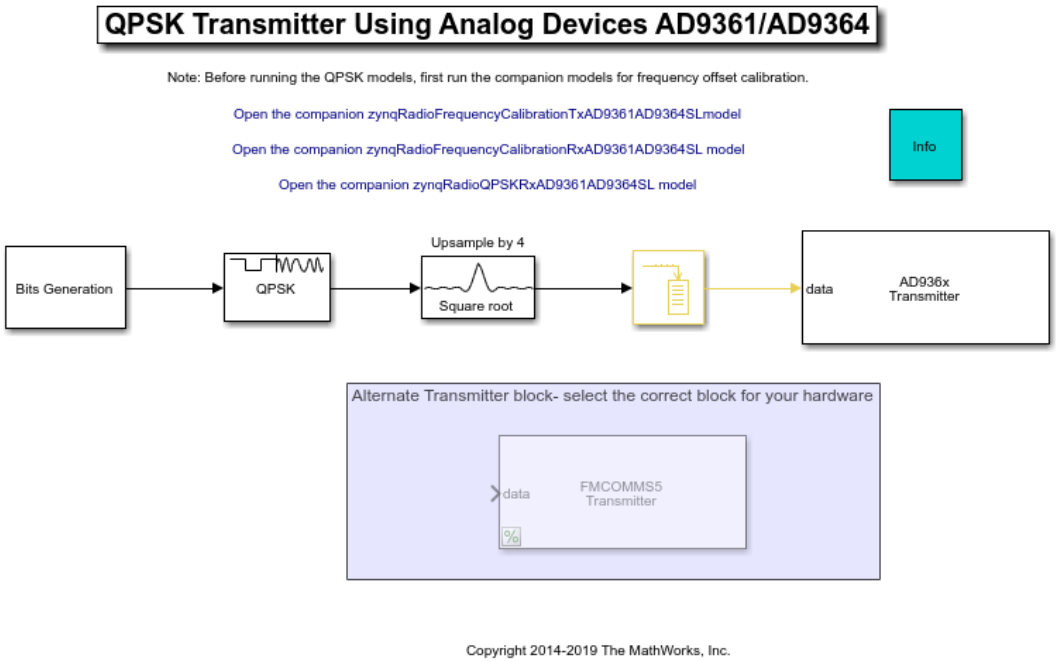


Figure 50. QPSK transmitter Simulink model. Source: [66].

Each frame contains a 26-bit header as a preamble. The first 105 bits of the message source is the ASCII representation of the indexed “Hello World ###” message which is matrix concatenated for a total of 174 bits. The 26-bit header is then matrix concatenated with the 174-bit message source to output a 200-bit frame for modulation. The bit generation process is shown in Figure 51.

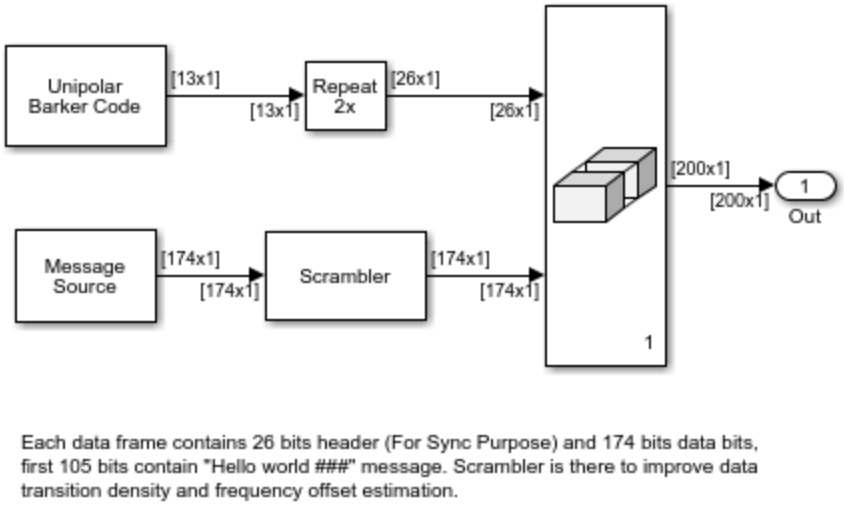


Figure 51. QPSK transmitter bit generation. Source: [66].

Parameters such as base band sample rate, up-sampling factor, and channel mapping can be changed through the QPSK transmitter initial parameters script, zynqRadioQPSKTxAD9361AD9364SL_init.m (Appendix D).

4. QPSK Receive Model

After the QPSK transmitter is run, the QPSK receiver is run and receives the signal from the transmitter, demodulates, and then decodes the message source to display “Hello world ###” in the diagnostics viewer. The QPSK receiver model is shown in Figure 52.

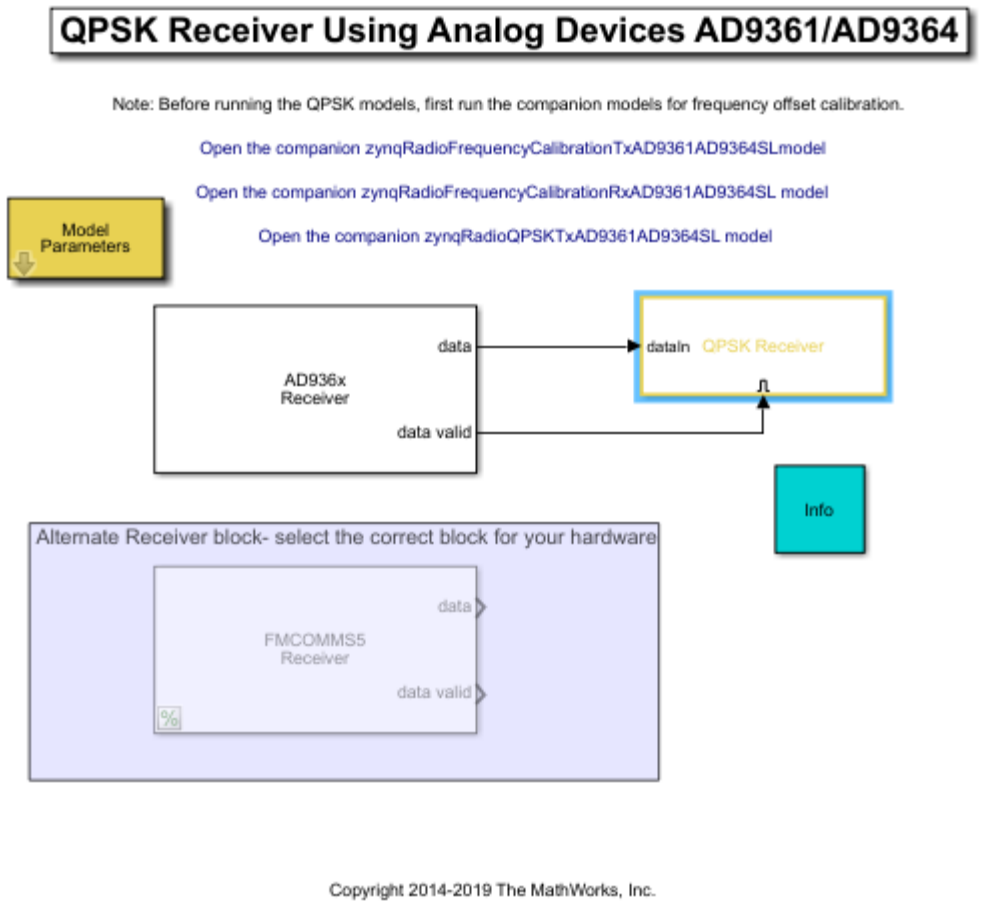


Figure 52. QPSK receiver Simulink model. Source: [67].

The QPSK receiver block conducts automatic gain control (AGC), raised cosine receive filter, course frequency compensation, fine frequency compensation, timing recovery, and data decoding, as shown in Figure 53.

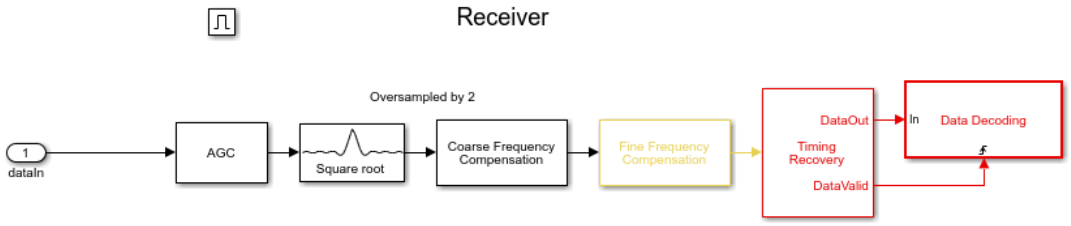


Figure 53. QPSK receiver block. Source: [67].

AGC applies a variable gain in order to keep the amplitude of the signal at 1/up-sampling factor. The raised cosine receive filter uses a roll-off factor of 0.5 to down-sample the input signal by two. The coarse frequency compensation estimates the frequency offset of the received signal and automatically corrects for offset using FFT. The fine frequency compensation uses a PLL on the residual frequency and phase offset not corrected by the coarse frequency compensation. Timing recovery then resamples the input signal and implements a PLL to correct for the timing error in the received signal. Finally, data decoding aligns the frame boundaries, resolves phase ambiguity created by the fine frequency compensation, demodulates the signal, and decodes the message source to display an indexed “Hello world ###” [67].

As was the case with the frequency offset calibration models, the baseband sampling rate must be identical in both the QPSK transmitter and receiver to successfully send and receive a signal. Parameters such as base band sample rate, up-sampling factor, and down-sampling factor can be changed through the QPSK receiver initial parameters script, zynqRadioQPSKRxAD9361AD9364SL_init.m (Appendix E).

V. TESTING AND RESULTS

The testing approach for this research followed three phases. The first step consisted of testing individual components to ensure proper operation and evaluate performance before integration. Second, components were integrated at a basic level to define interfaces and measure expected output. Finally, the components of the X-band SDR payload were integrated for a functional end-to-end bench test to test for transmission data rate, QPSK modulation/demodulation, and size and weight limitations imposed by the radio assembly mechanical enclosure.

A. COMPONENT TESTING

1. SoM

The ADRV9361-Z7035 SoM was mounted to the custom carrier board, powered by 5V, and connected to an input PC through ethernet to a Linksys USB3GIG USB-Ethernet adapter. A U.FL-to-U.FL coaxial cable connected the TX1A transmit port to the RX1A receive port in a loopback configuration. This setup of this component-level test is shown in Figure 54.

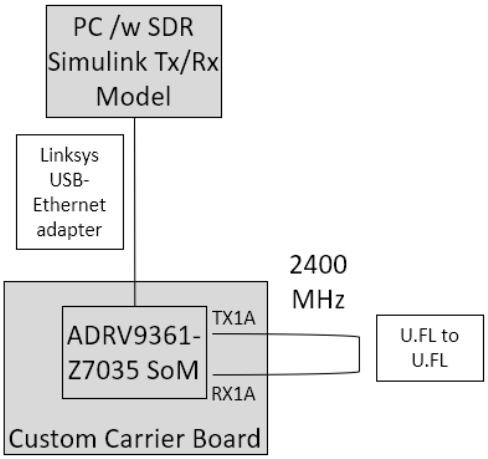


Figure 54. SoM component testing.

The frequency offset calibration transmit and receive models were run on the input PC to determine and correct for the frequency offset. For this component-level test, the center frequency was left at the default 2.4 GHz and the baseband sampling rate was left at the default 520.841 kHz, as shown in Figure 55, Figure 56, and Figure 57.

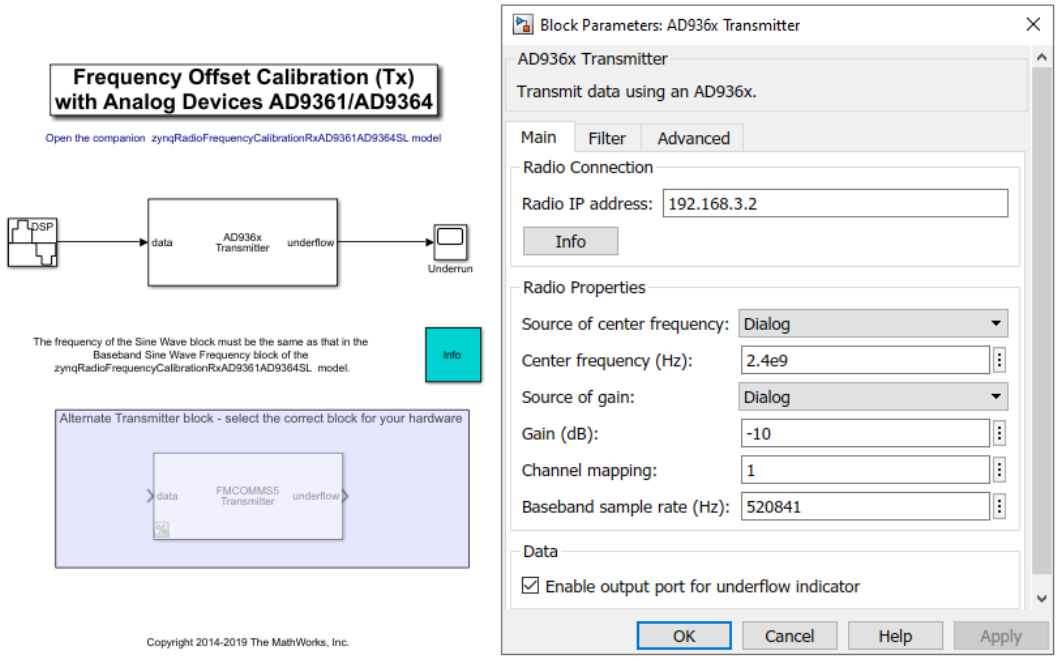


Figure 55. Transmit frequency offset calibration.

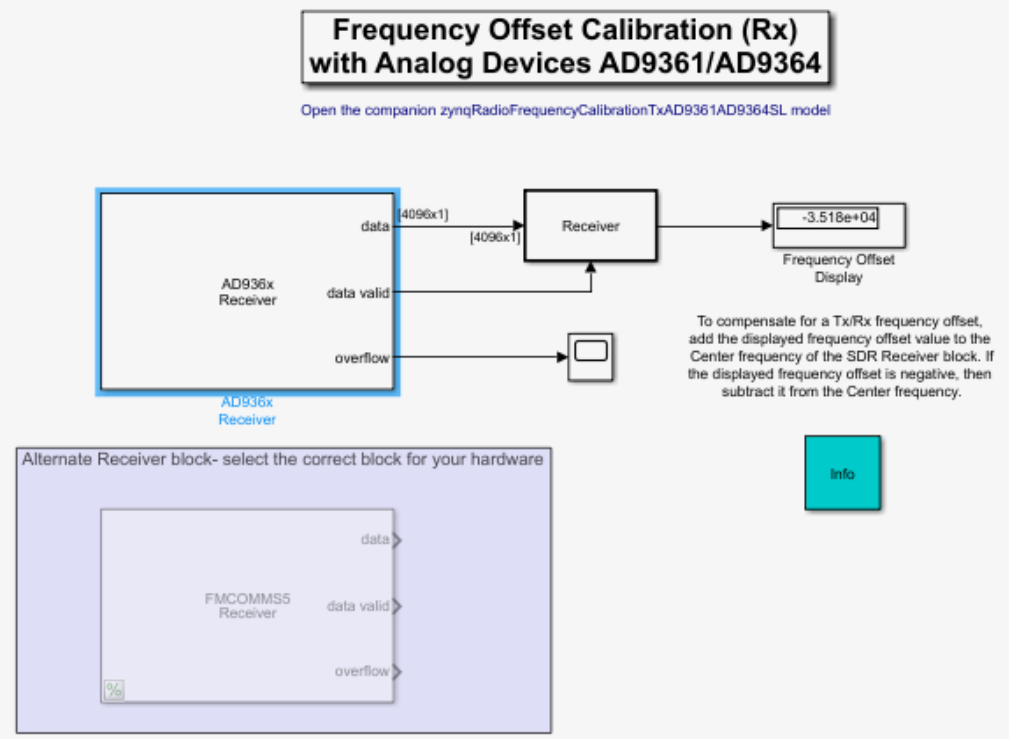


Figure 56. Receiver frequency offset calibration display.

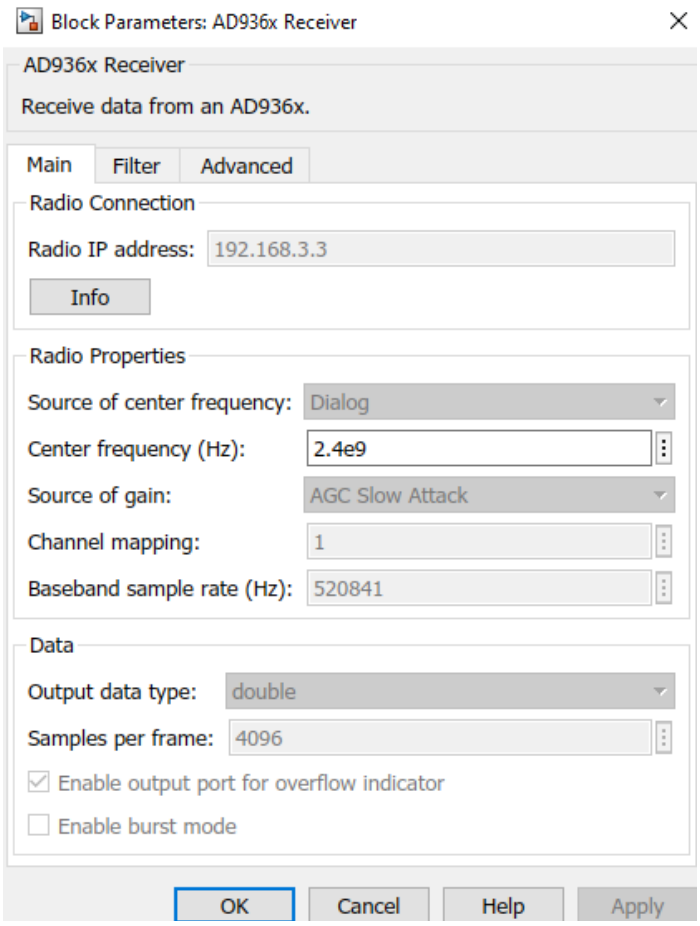


Figure 57. Receiver frequency offset calibration block parameters.

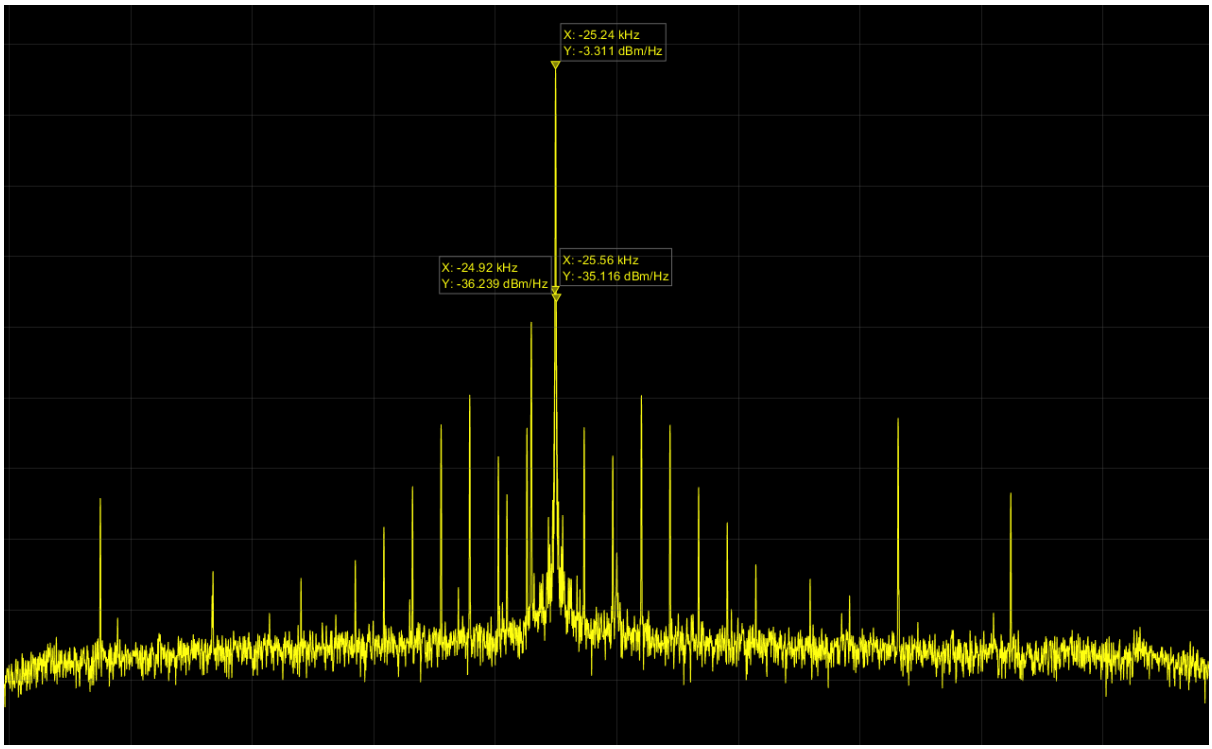


Figure 58. Spectrum analyzer output for frequency offset calibration.

From the frequency offset display of the receive model in conjunction with the spectrum analyzer output in Figure 58, we can see that the frequency offset is approximately -3.53e4 Hz. Adding this offset back to the center frequency of the receive model, we can drive the offset to near zero with a residual of approximately 45.52 Hz as shown in Figure 59 and Figure 60. The 10 kHz transmitted tonal is then received as 10.05 kHz, shown in Figure 61.

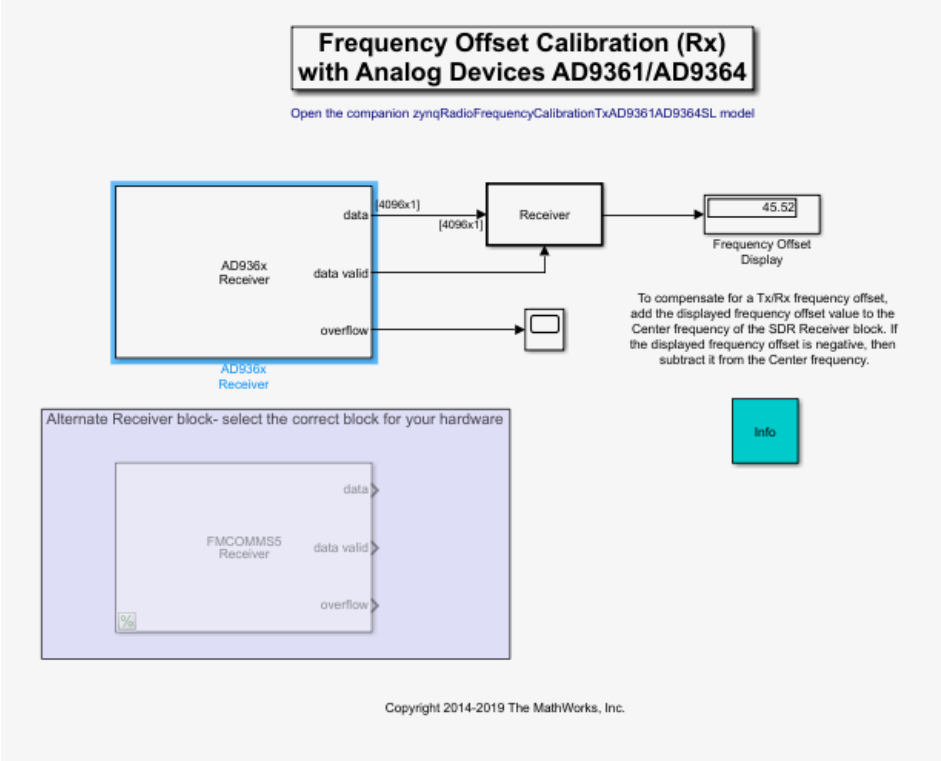


Figure 59. Frequency offset added into receive calibration model.

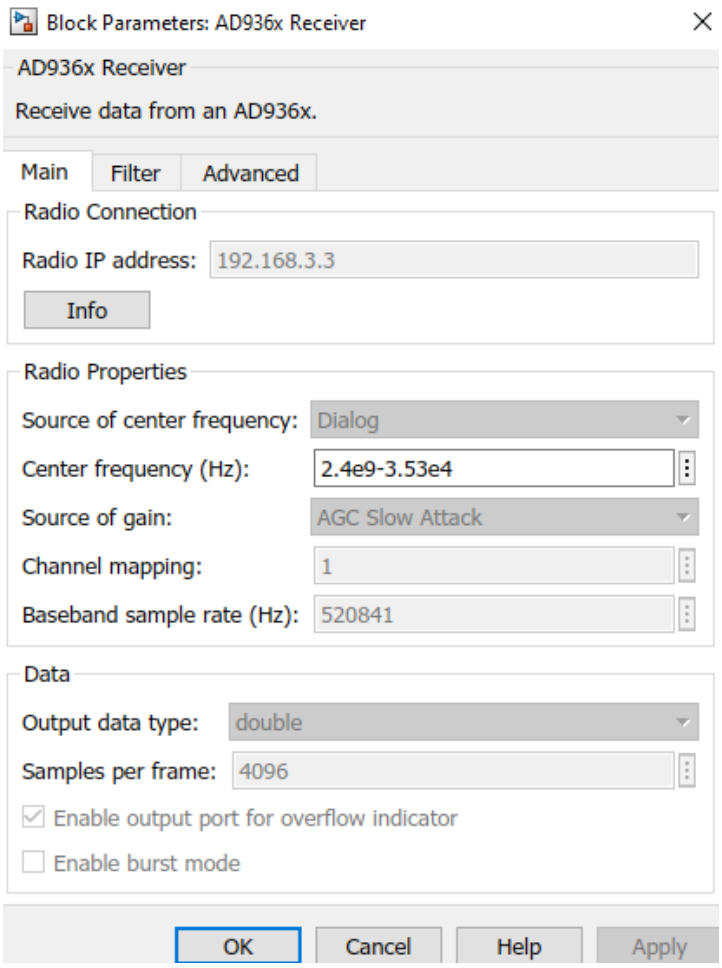


Figure 60. Frequency offset added in block parameters.

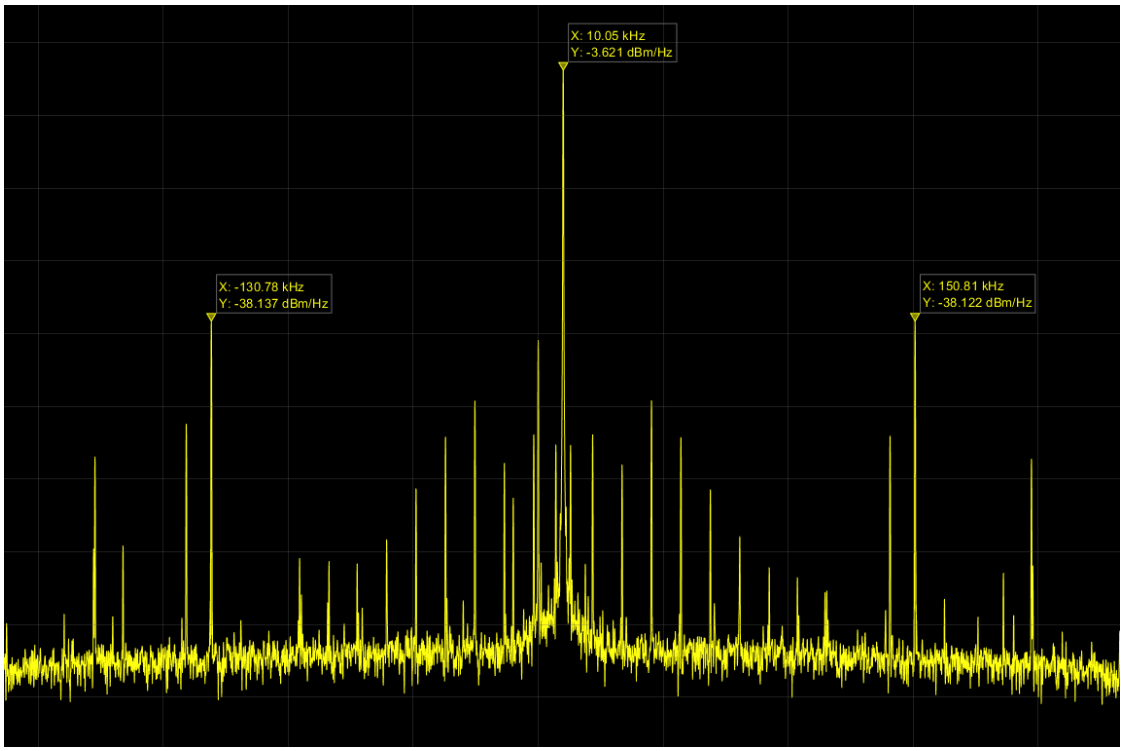


Figure 61. Spectrum analyzer with offset added into receive calibration model.

Once the system was calibrated for frequency offset, the author ran the QPSK transmitter and receiver models with this same frequency offset added to the center frequency parameter of the receive model, as shown in Figure 62, Figure 63, Figure 64, and Figure 65.

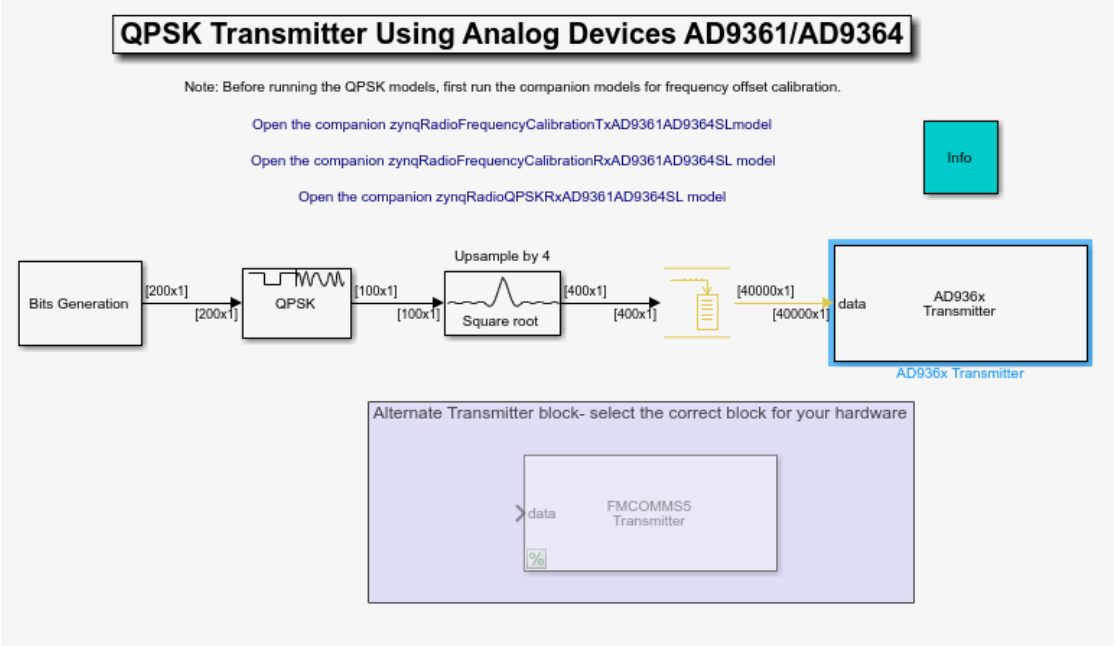


Figure 62. QPSK transmitter model.

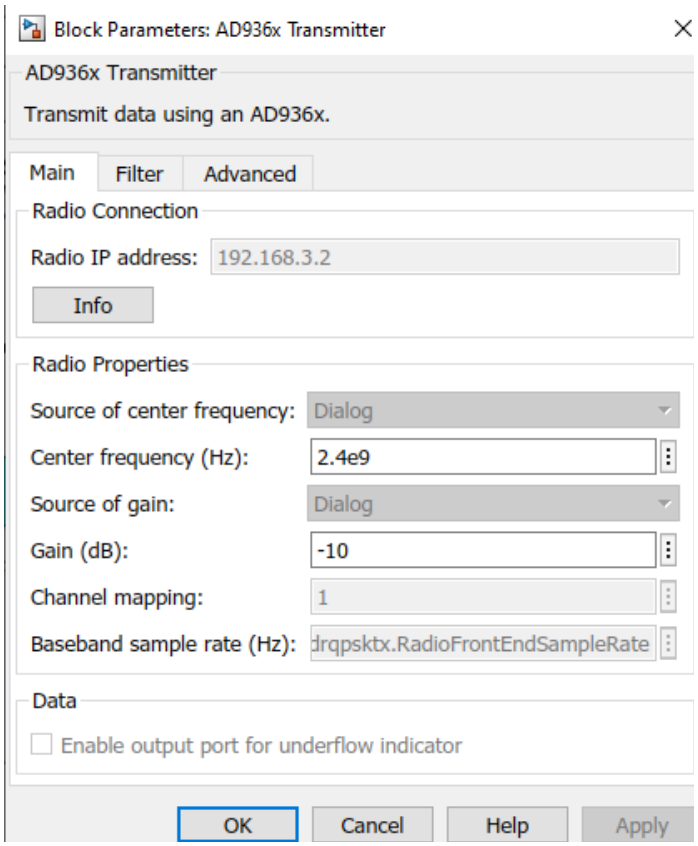


Figure 63. QPSK transmitter block parameters.

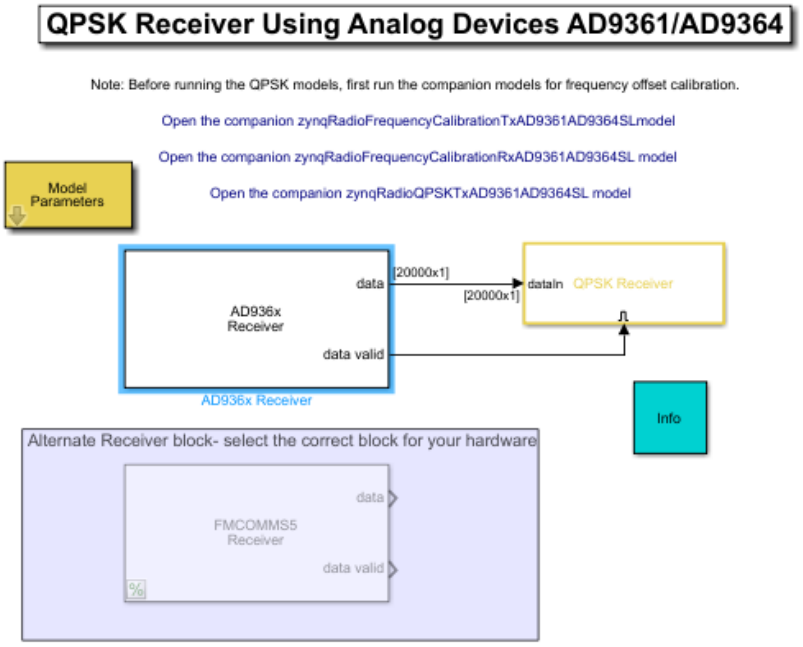


Figure 64. QPSK receiver model.

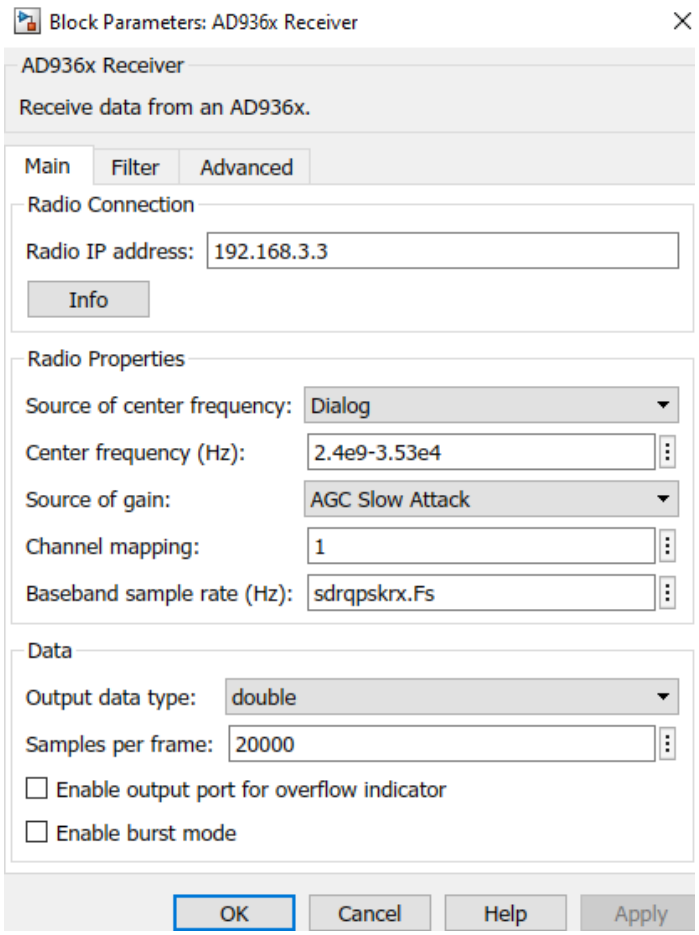


Figure 65. QPSK receiver block parameters.

Figure 66 shows the receiver successfully demodulating and decoding the message source to display an indexed “Hello world ###” in the diagnostics viewer window, indicating a successful loopback test of the SoM.

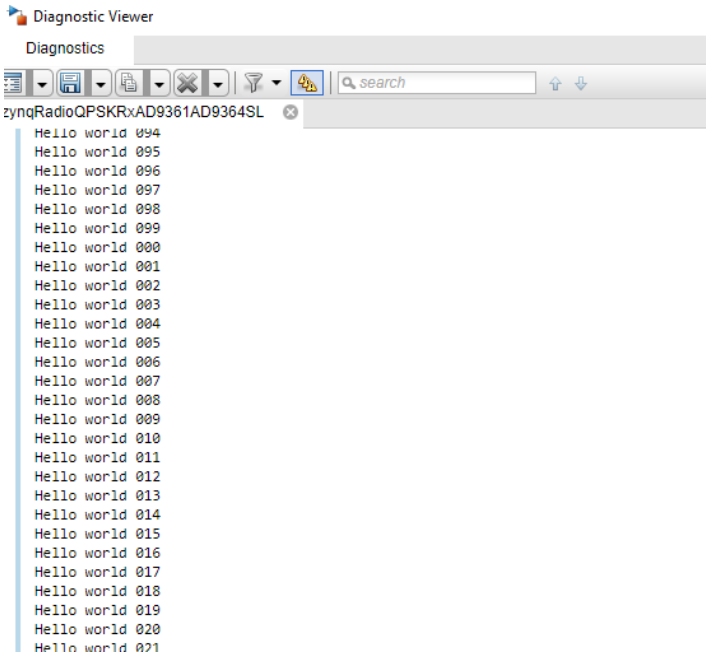


Figure 66. QPSK receiver output.

While the models were still running, the loopback cable was removed and the transmit port TX1A was connected to the spectrum analyzer to verify the output signal. During this test, it was found that the receiver model still successfully decoded most of the transmitted message source with only a few missed symbols where the “Hello world ###” message was not displayed clearly. This drove the requirement to use an independent SDR to receive the signal during integrated and end-to-end functional testing, rather than continue testing via the loopback configuration.

2. Convert Board

The convert board was powered by 5V from an Agilent E362A DC power supply, drawing 0.416A of current. A Hewlett Packard 8648C signal generator provided a 2.5125 GHz reference signal over coaxial cable to the SMA input of the convert board in order to mimic the input it would eventually receive from the SDR. The output was connected to a N9915A FieldFox spectrum analyzer via coaxial cable with an estimated loss of 0.5 dB. This configuration is shown in Figure 67.

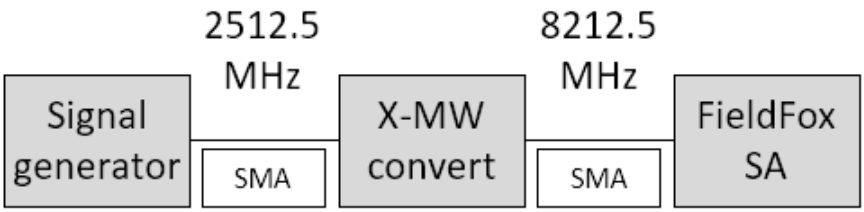


Figure 67. Convert board component testing configuration.

The signal received by the spectrum analyzer was at the expected 8.2125 GHz frequency, as shown in Figure 68. This indicated a successful up-conversion and verified component testing of the convert board.

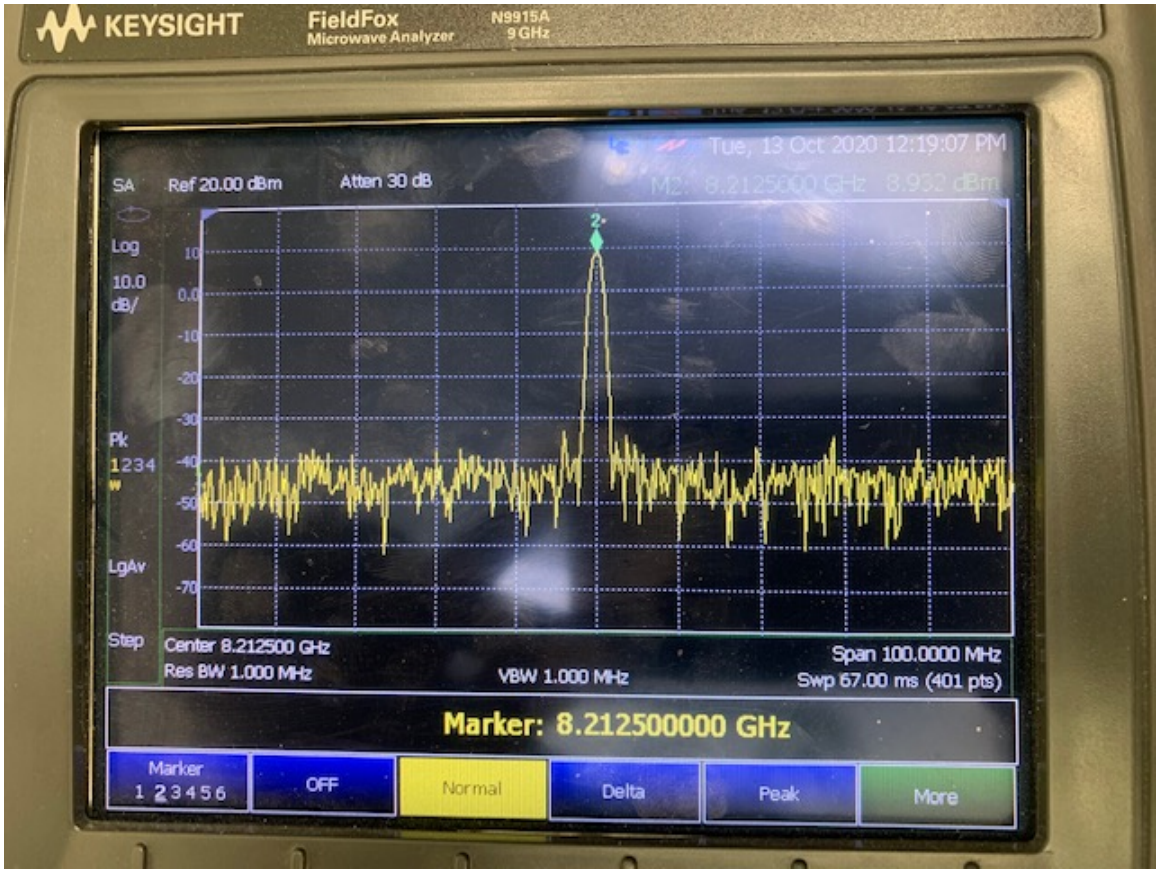


Figure 68. Convert board component testing.

3. X-band PLL LNB

The AD9361 integrated RF agile transceiver is constrained by a tuning range of 70 MHz to 6 GHz. Since our convert board shifts the 2.5125 GHz signal transmitted by the SDR to 8.2125 GHz, it is necessary to down-convert the signal to be successfully received. This is accomplished through an X-band PLL low-noise block (LNB) downconverter from Swedish Microwave. The downconverter, shown in Figure 69, has a frequency range of 7.75 to 8.50 GHz, features a 6.8 GHz LO and is powered by 12 to 24 V-DC.

Figure 69. X-band PLL LNB. Source: [68].

The LNB adds +60 dBm of attenuation and as such, necessitates -60 dBm of attenuation on the input to net 0 dBm and reduce the SNR of the system. For this component testing, the FieldFox spectrum analyzer sent a reference signal of 8.2125 GHz through a -60 dBm attenuator to the LNB powered by 12 V. After being down-converted by 6.8 GHz, the expected signal output of 1.4125 GHz was measured by the spectrum analyzer. This configuration is shown in Figure 70. The spectrum analyzer successfully received the 1.4125 GHz output and identified the transfer function of the LNB, as shown in Figure 71.

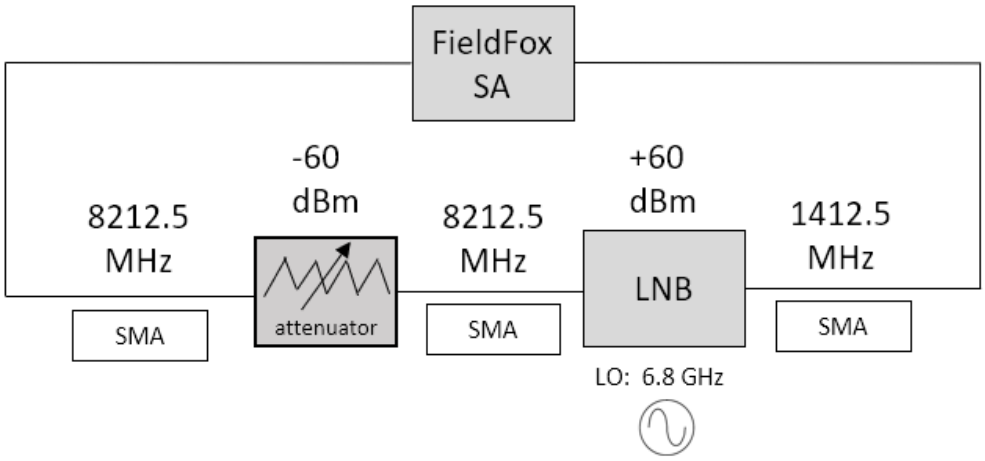


Figure 70. LNB component testing configuration.

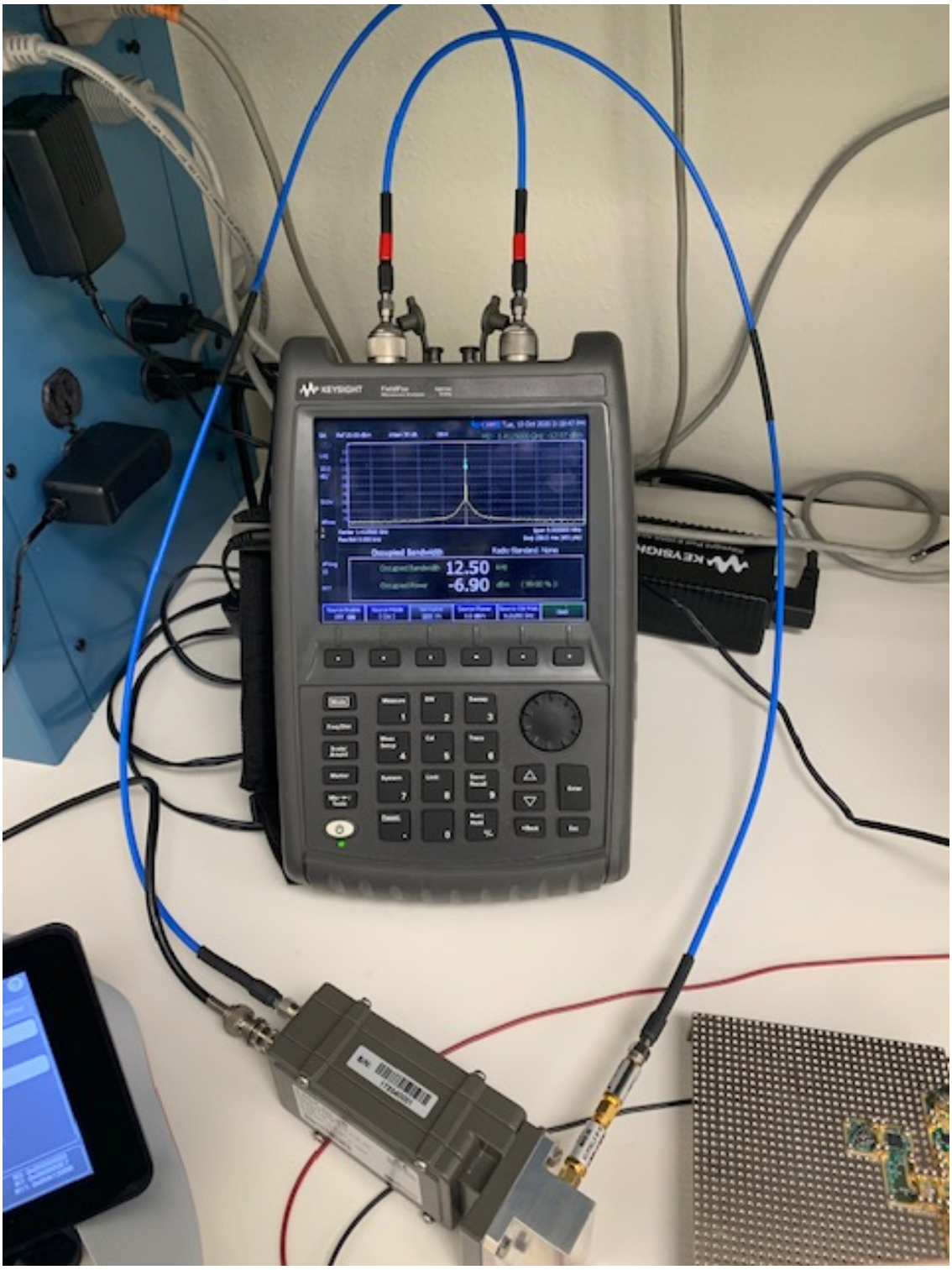


Figure 71. LNB component testing.

B. INTEGRATED TESTING

Following successful component testing, the author began to integrate components to measure performance and prepare for end-to-end functional testing.

1. SoM to Convert Board

Integrated testing began with the first half of the X-band SDR components. The author verified that the SoM was able to interface with the convert board and that the convert board would successfully upconvert the signal output from the SDR to the expected 8.2125 GHz X-band transmission. Here, a host PC ran the QPSK transmitter model to operate the SDR at 2.5125 GHz. A U.FL-to-SMA coaxial cable connected the output of the SDR to the input of the convert board, while the output of the convert board was connected to the FieldFox spectrum analyzer. The configuration for this portion of the integrated test is shown in Figure 72.

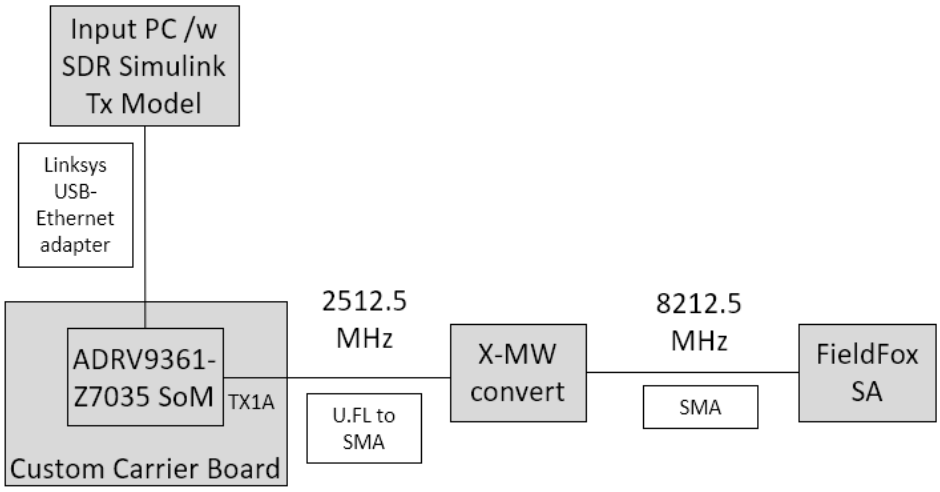


Figure 72. SoM to convert board integrated testing configuration.

Screen captures of the spectrum analyzer show that the convert board successfully up-converted the 2.5125 GHz IF to 8.2125 GHz and identified the occupied bandwidth of 1.10 MHz and occupied power level at -5.65 dBm of the signal, as seen in Figure 73 and

Figure 74. These characteristics were expected for a QPSK transmission in the X-band frequency range.

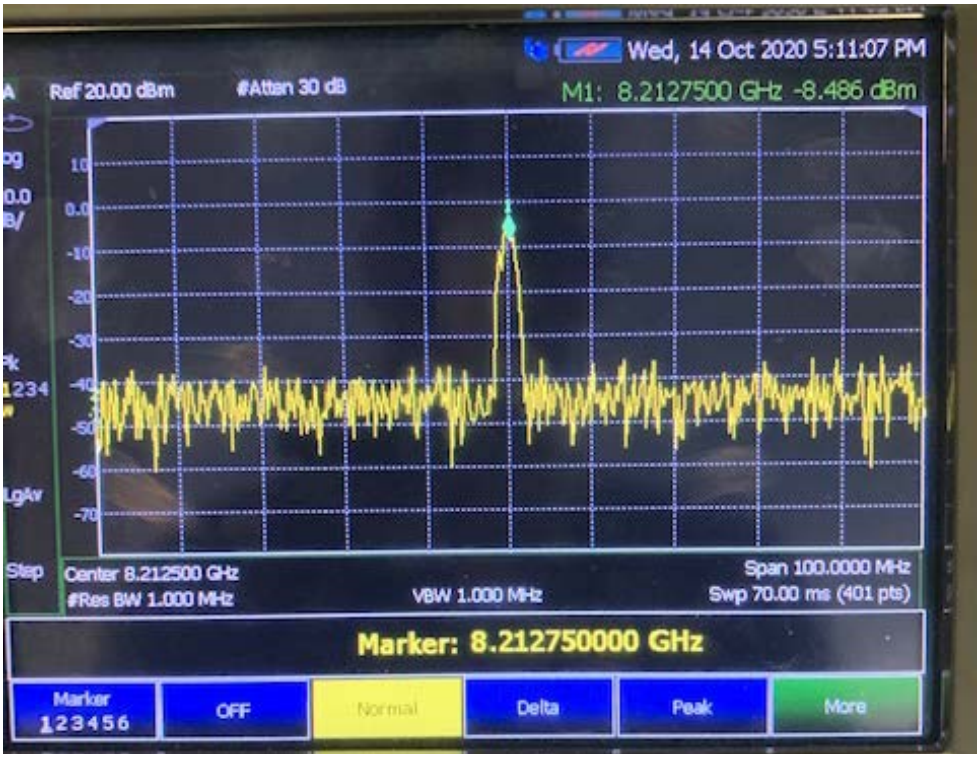


Figure 73. FieldFox SA output.

Figure 74. FieldFox SA occupied bandwidth.

2. SoM to ZedBoard

Component testing identified the need for a separate SDR to receive as opposed to conducting testing through loopback alone. To accomplish this, the author used the initial hardware design of the ZedBoard with ADFMComms3-EBZ SDR mounted via FMC-LPC and operated through a second host PC, to receive the QPSK signal from the SoM. Integrated testing was executed by running the QPSK transmitter model on the primary PC with the ADRV9361-Z7035 while the secondary PC with the ZedBoard/ADFMComms3-EBZ SDR ran the QPSK receiver model. The configuration for this integrated testing is show in Figure 75.

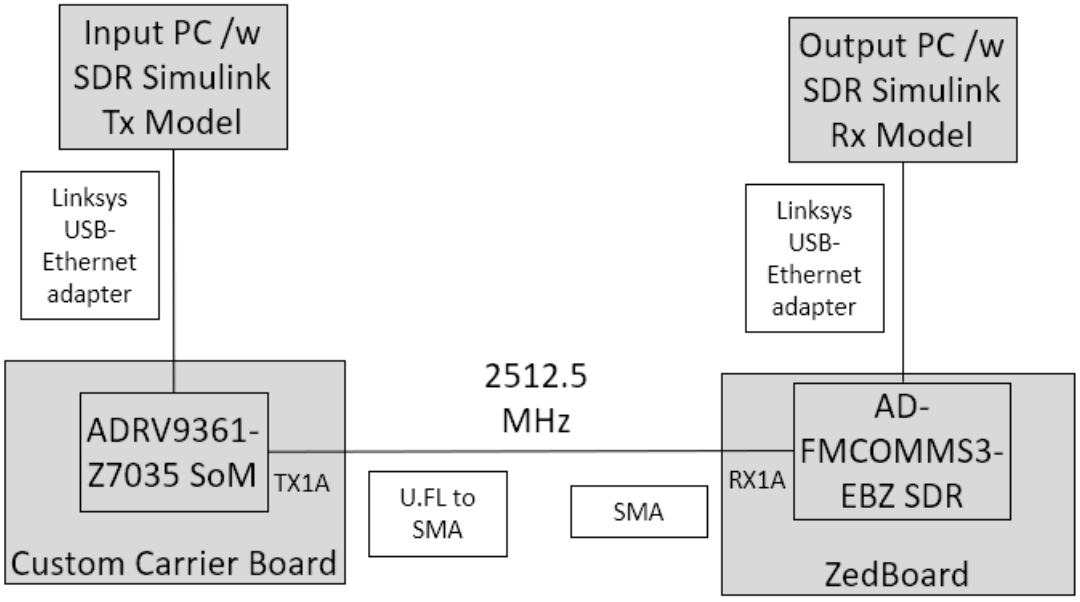


Figure 75. SoM to ZedBoard integrated testing.

For this integrated testing, the center frequency of both the transmit and receive models were set to 2.5125 GHz, as the convert board was not connected, and the baseband sampling rate was left at the default 520.841 kHz. Before the QPSK transmit and receive models were run, frequency offset calibration models were executed, as seen in Figure 76, Figure 77, and Figure 78.

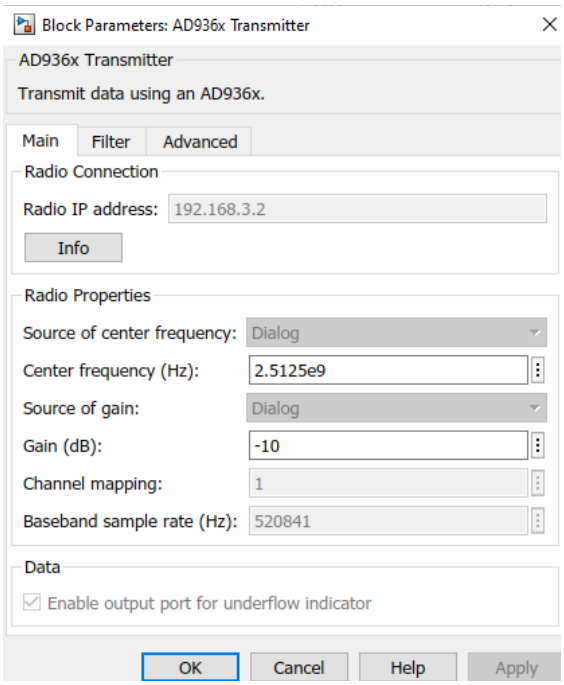


Figure 76. SoM to ZedBoard transmit frequency offset calibration block parameters.

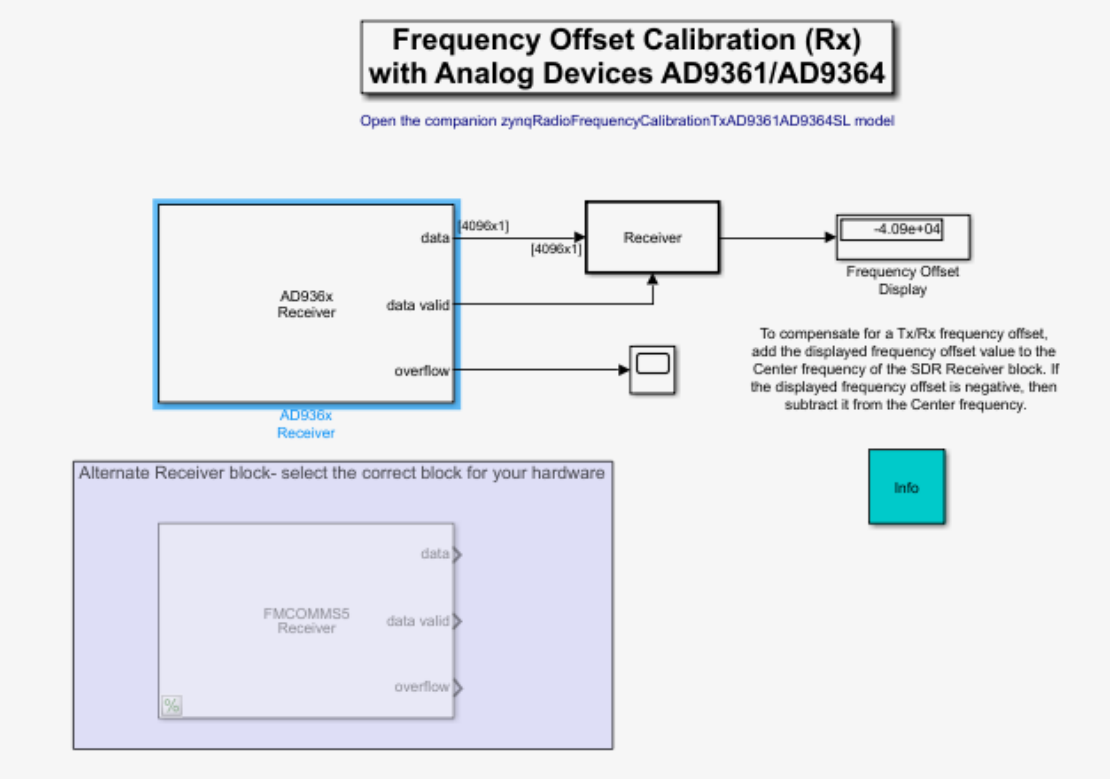


Figure 77. SoM to ZedBoard receive frequency offset calibration model.

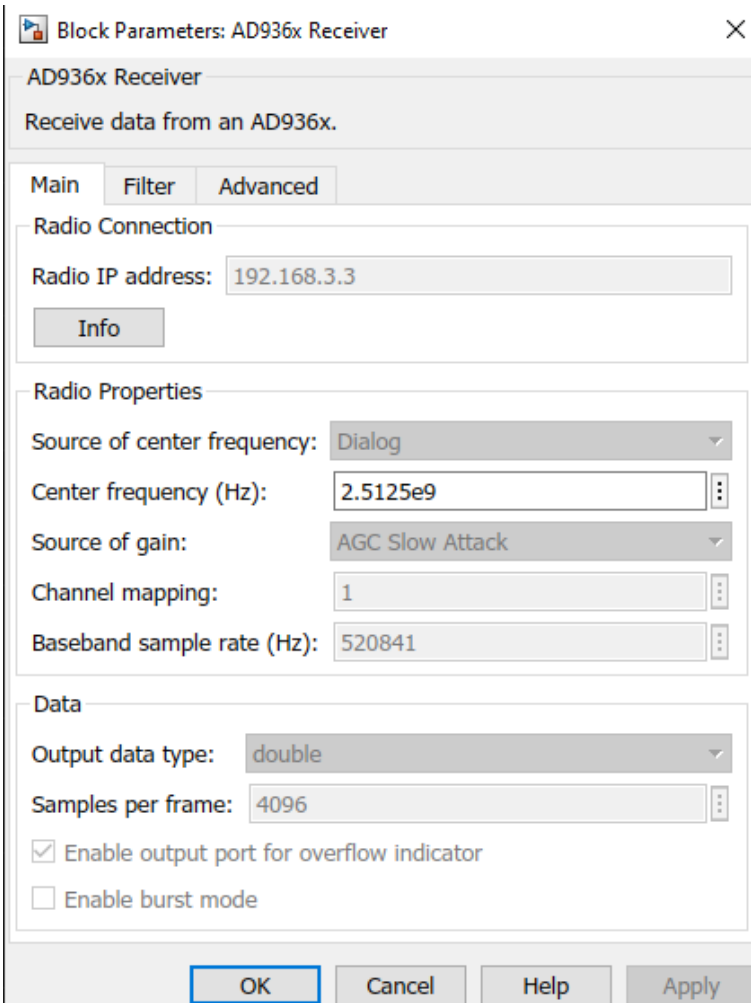


Figure 78. SoM to ZedBoard receive frequency offset calibration block parameters.

From the frequency offset display and spectrum analyzer of the receive model, shown in Figure 79, the frequency offset was determined to be approximately 4.1e4 Hz.

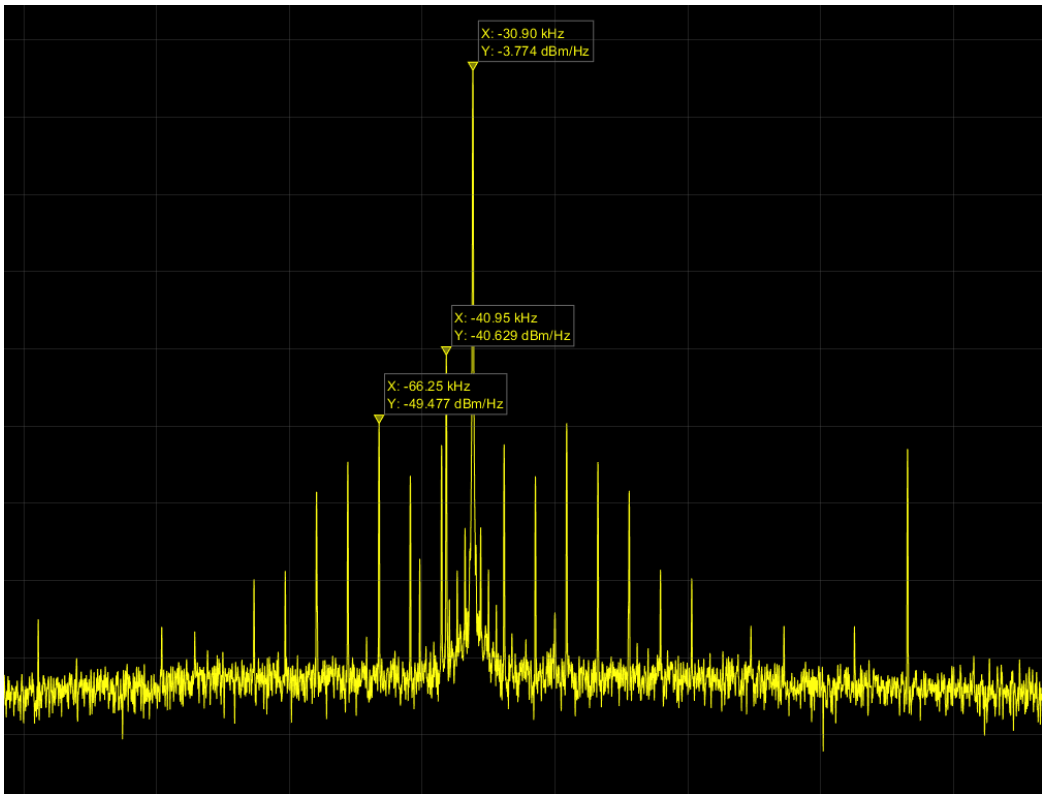


Figure 79. Spectrum analyzer frequency offset calibration.

This offset was added to the center frequency parameter of the receive model, as seen in Figure 80 and Figure 81, and the calibration model was run again to drive the offset frequency to near zero, as seen in Figure 82.

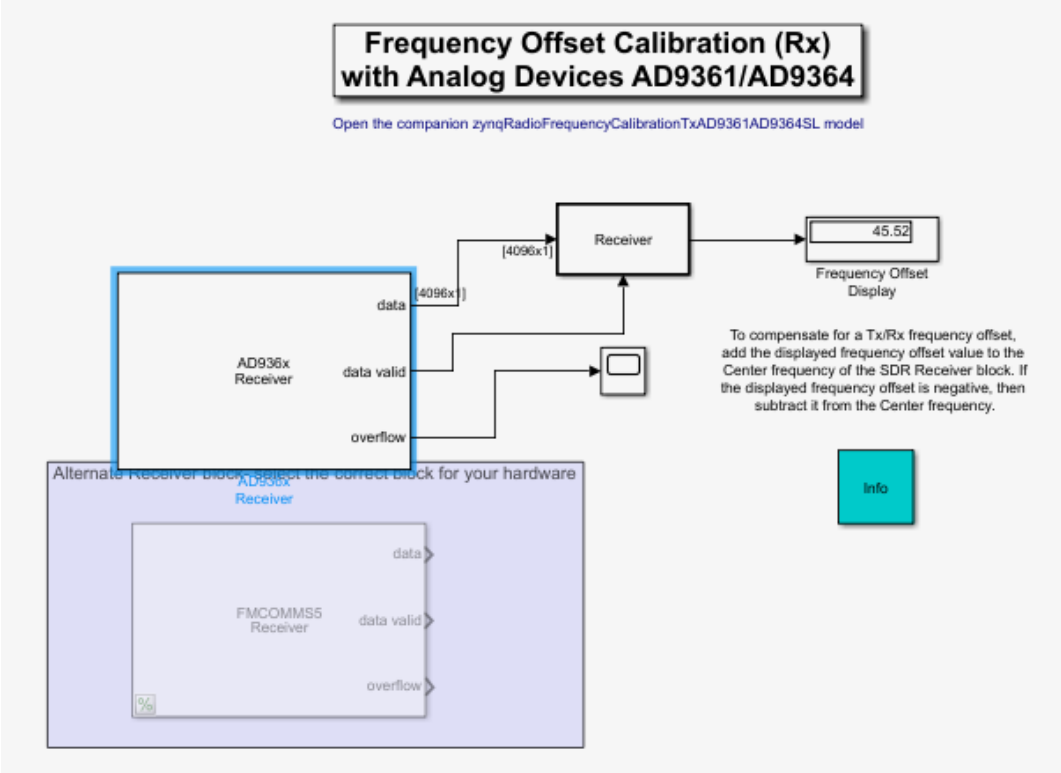


Figure 80. Tuned frequency offset calibration display of receiver.

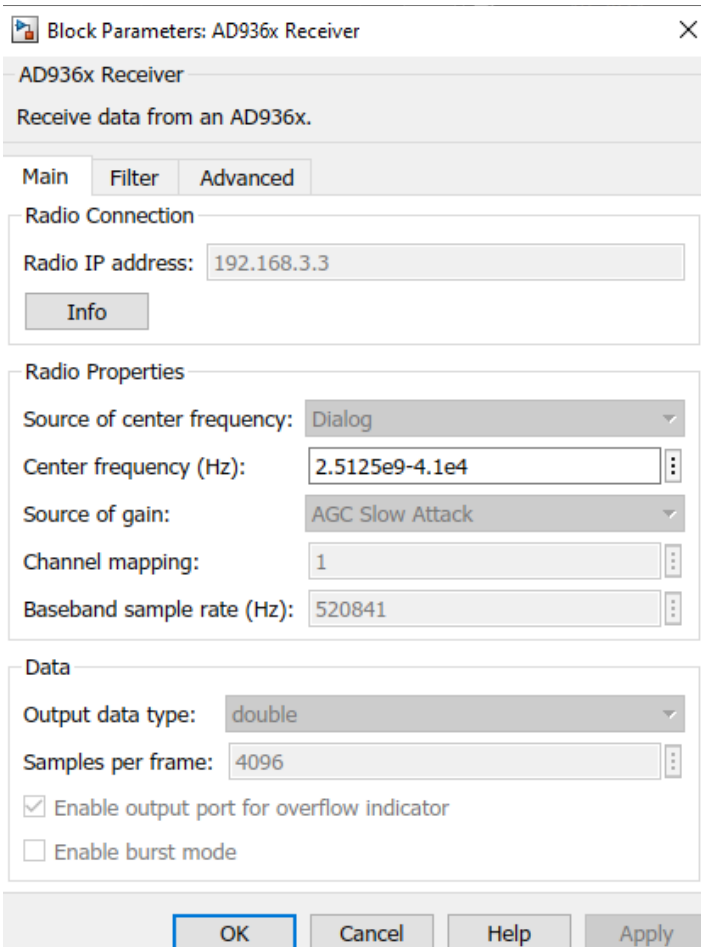


Figure 81. Tuned frequency offset calibration receiver block parameters.

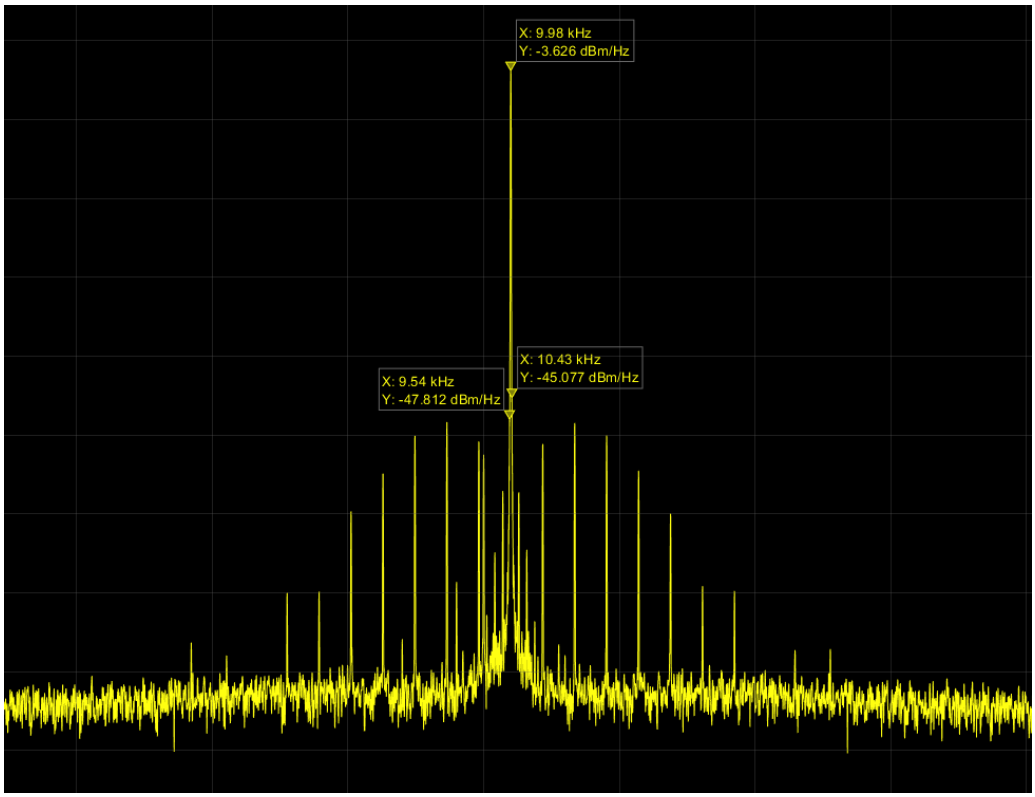


Figure 82. Tuned spectrum analyzer frequency offset calibration.

This same frequency offset was added to the center frequency of the QPSK receiver on the secondary PC. The QPSK transmitter model was executed first on the primary PC, seen in Figure 83, followed by the calibrated QPSK receiver model on the secondary PC with added frequency offset shown in Figure 84.

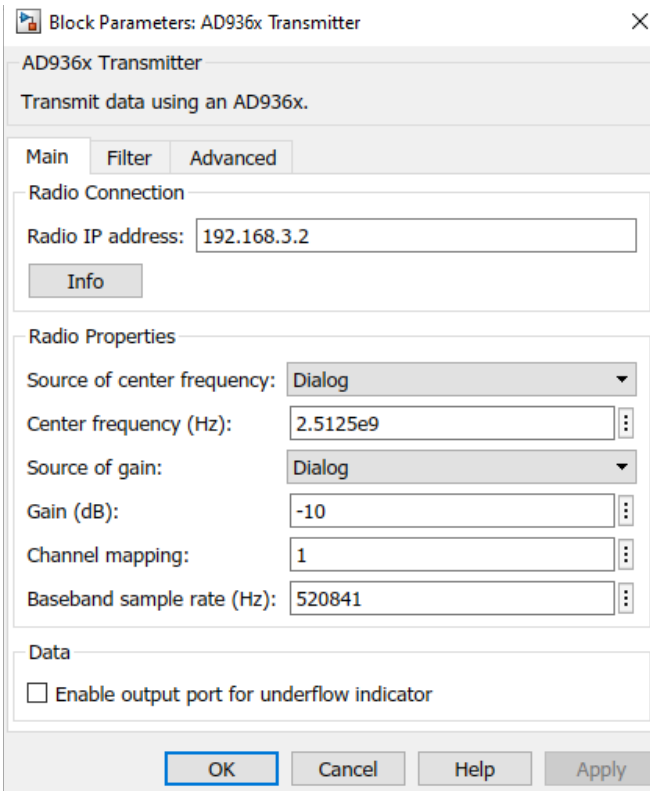


Figure 83. SoM to ZedBoard integrated testing QPSK transmitter block parameters.

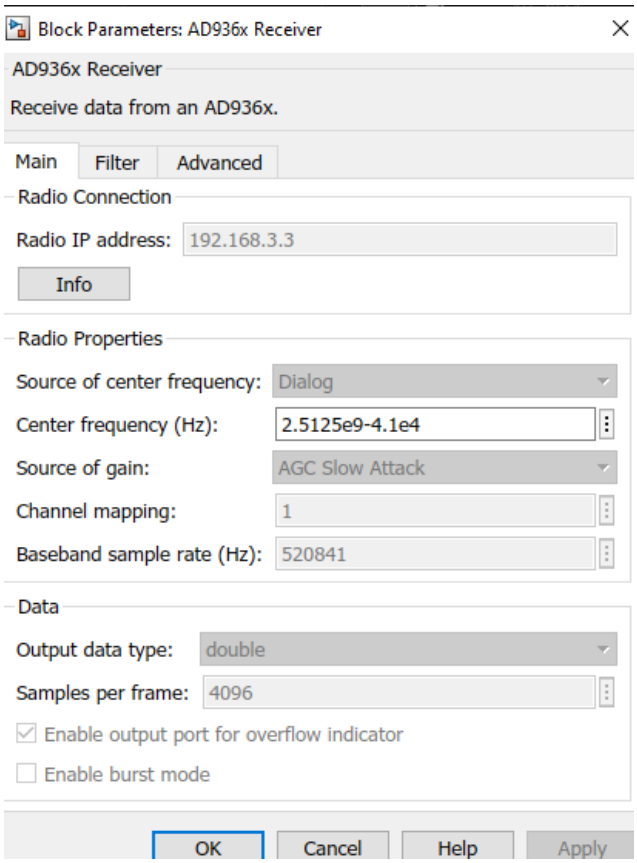


Figure 84. SoM to ZedBoard integrated testing QPSK receiver.

Figure 85 shows the indexed “Hello world ###” decoded message source from the diagnostics viewer of the receiver model on the secondary PC, indicating a successful QPSK modulation/demodulation through the integrated testing, from the SoM to the ZedBoard with ADFMComms3-EBZ SDR.

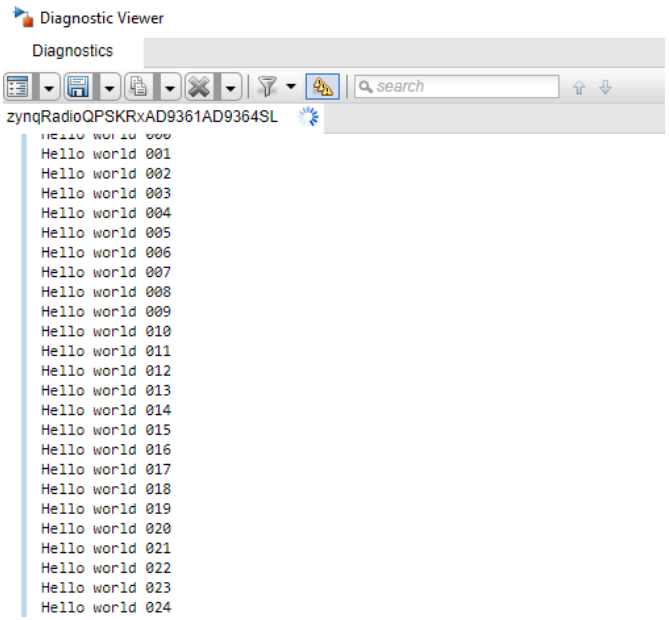


Figure 85. Integrated testing successful QPSK modulation/demodulation and decoding from SoM to ZedBoard/FMComms3-EBZ SDR.

Screen captures of the FieldFox spectrum analyzer show the signal characteristics and occupied bandwidth of the output signal from the SoM, shown in Figure 86 and Figure 87. In comparison with the signal characteristics of the X-band transmission seen in the integrated testing of the convert board, this S-band transmission has a smaller occupied bandwidth and output power as is expected.

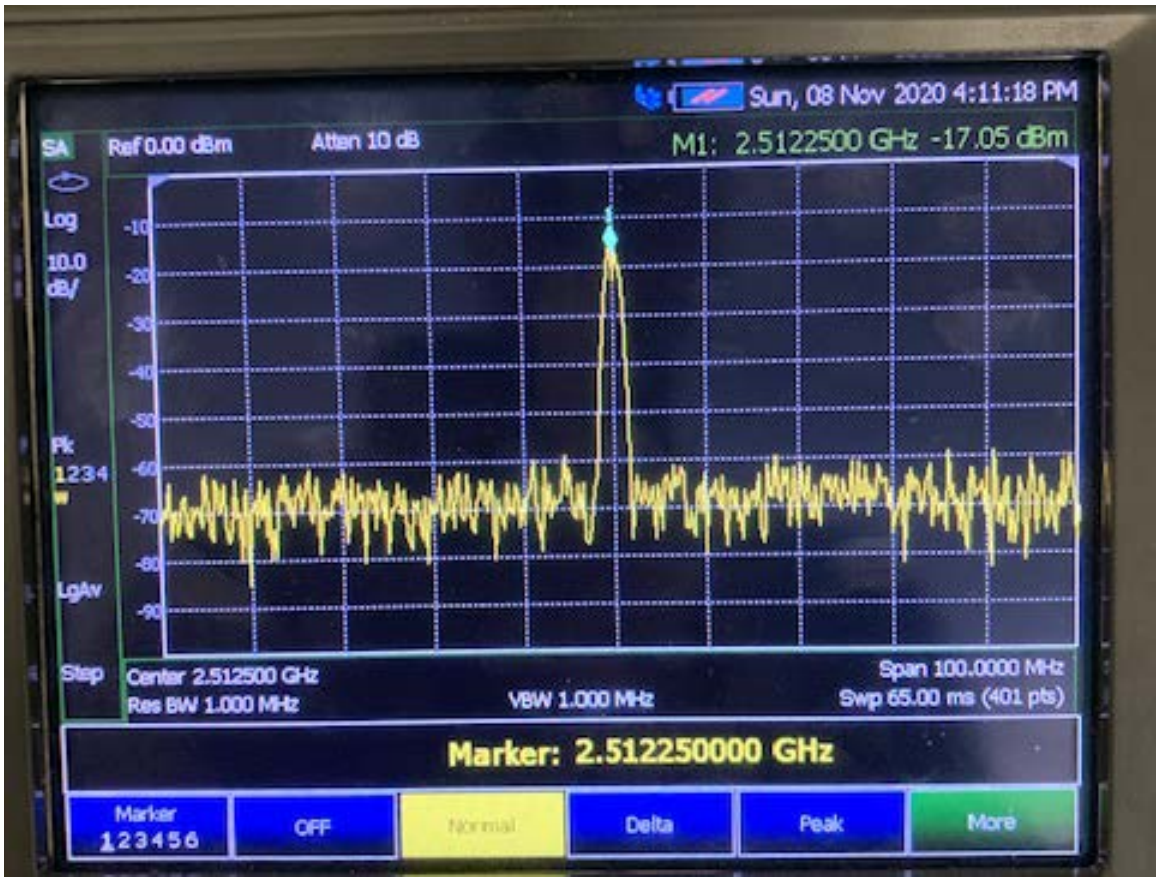


Figure 86. QPSK signal from SoM to ZedBoard/ADFMComms3-EBZ SDR.

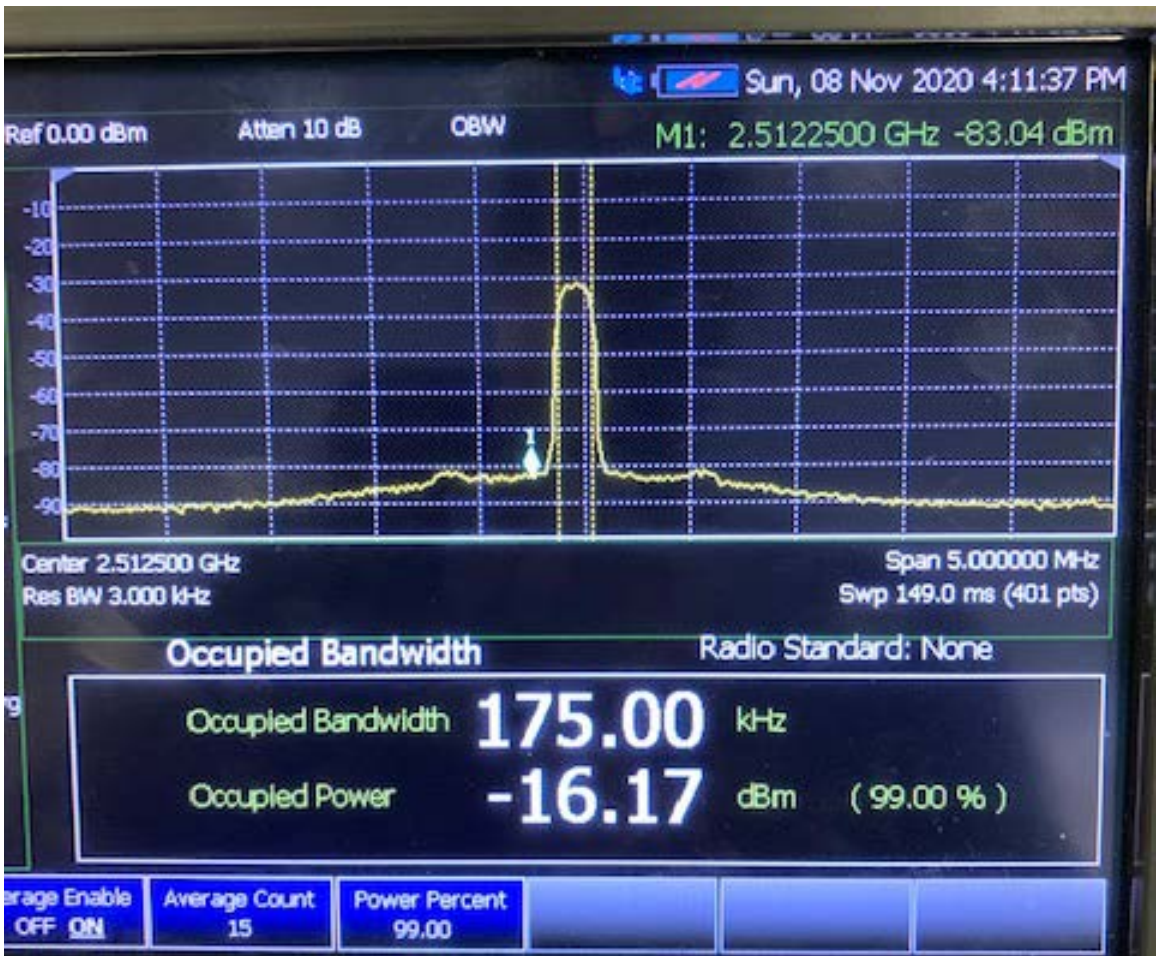


Figure 87. Occupied bandwidth of the QPSK signal from SoM to ZedBoard/ADFMComms3-EBZ SDR.

C. FUNCTIONAL TESTING

Functional end-to-end testing sought to satisfy the requirements of this research for high data rate transmission utilizing QPSK modulation. Following successful component testing and integrated testing, the end-to-end X-band SDR system was linked to conduct functional bench testing. A primary PC operated the SoM and ran the QPSK transmitter model at 2.5125 GHz. This output was connected from the TX1A U.FL port of the SoM to the convert board via coaxial cable. The upconverted 8.2125 GHz output of the convert board was run through coaxial cable to a -60 dBm attenuator before entering the X-band PLL LNB at +60 dBm. The 1.4125 GHz output of the LNB was then passed via coaxial cable to the ADFMComms3-EBZ SDR mounted on the ZedBoard, and received through

the QPSK receiver model running on a secondary PC. The configuration for this testing can be seen in Figure 88.

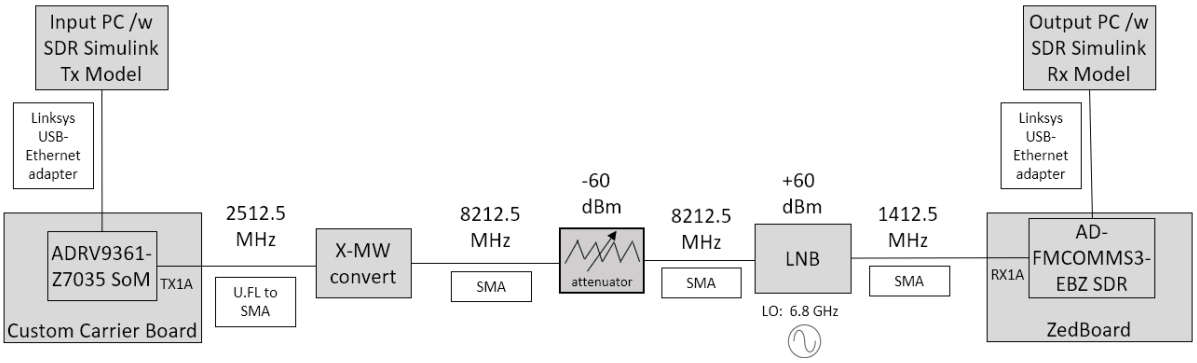


Figure 88. Functional end-to-end bench testing configuration.

Before the QPSK transmitter and receiver models could be run, frequency offset calibration was required to tune the receiver for the system. The calibration transmitter was run on the primary PC with a center frequency of 2.5125 GHz, while the calibration receiver was run on a secondary PC with a center frequency of 1.4125 GHz to match the output of the signal from the LNB. The baseband sample rate, which controls the data rate, was left at the default 520.841 kHz for this phase of testing. The calibration models, necessary to be run given the change in frequency paraters for the receive model as well as the addition of all payload components, can be seen in Figure 89, Figure 90, and Figure 91.

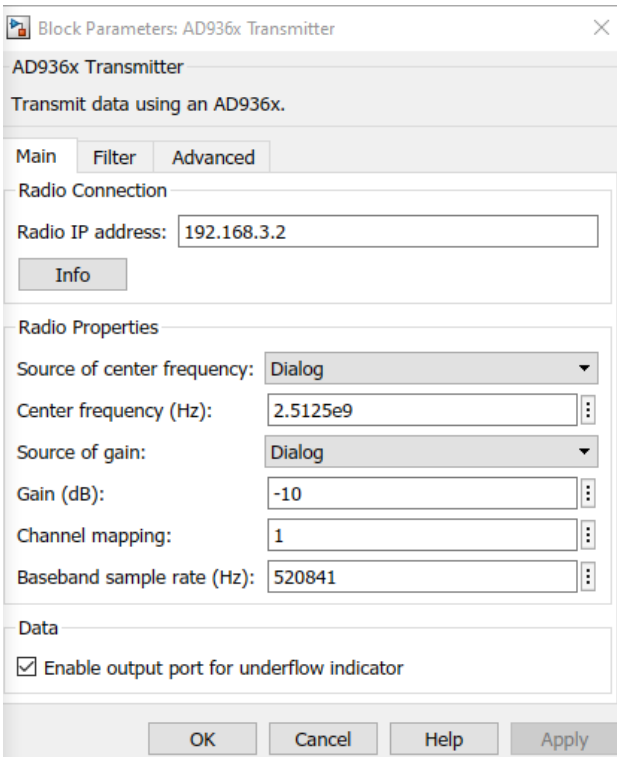


Figure 89. Functional testing frequency offset calibration transmitter.

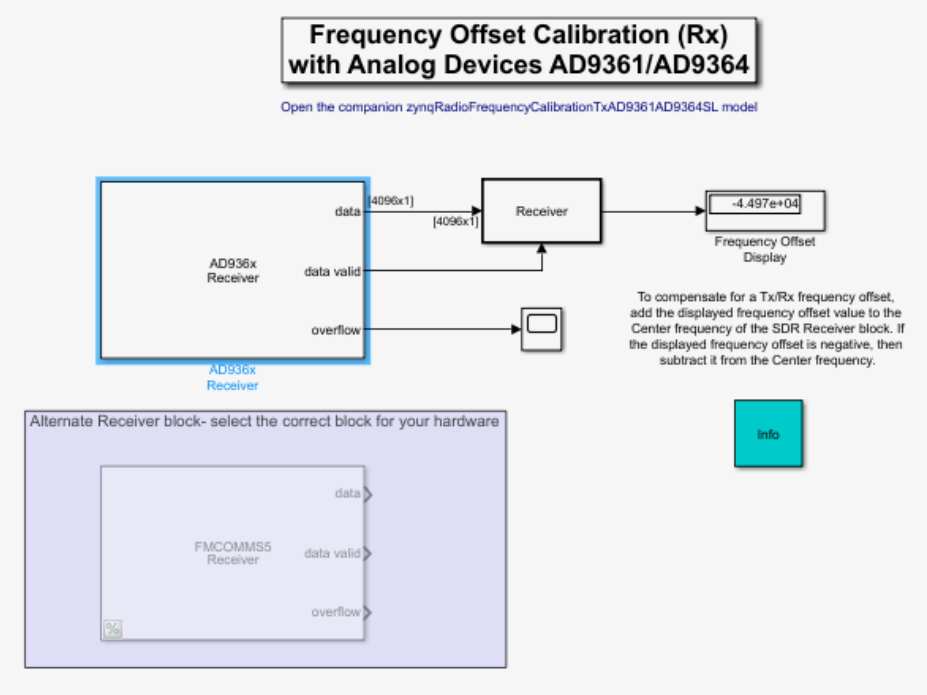


Figure 90. Functional testing frequency offset calibration receiver model.

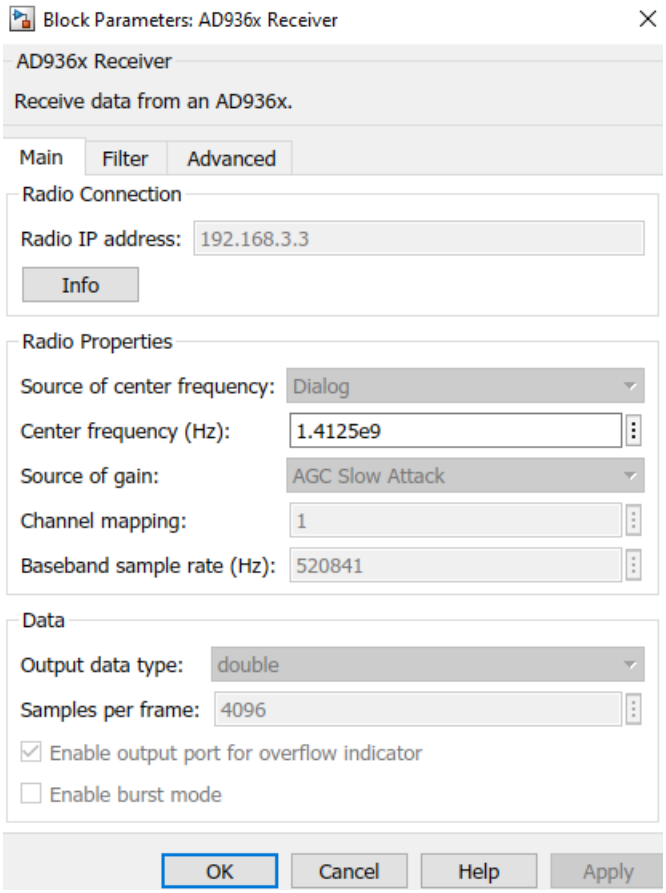


Figure 91. Functional testing frequency offset calibration receiver block parameters.

From the frequency offset display and spectrum analyzer output of the receiver model shown in Figure 92, the frequency offset was estimated to be approximately -4.515e4 Hz.

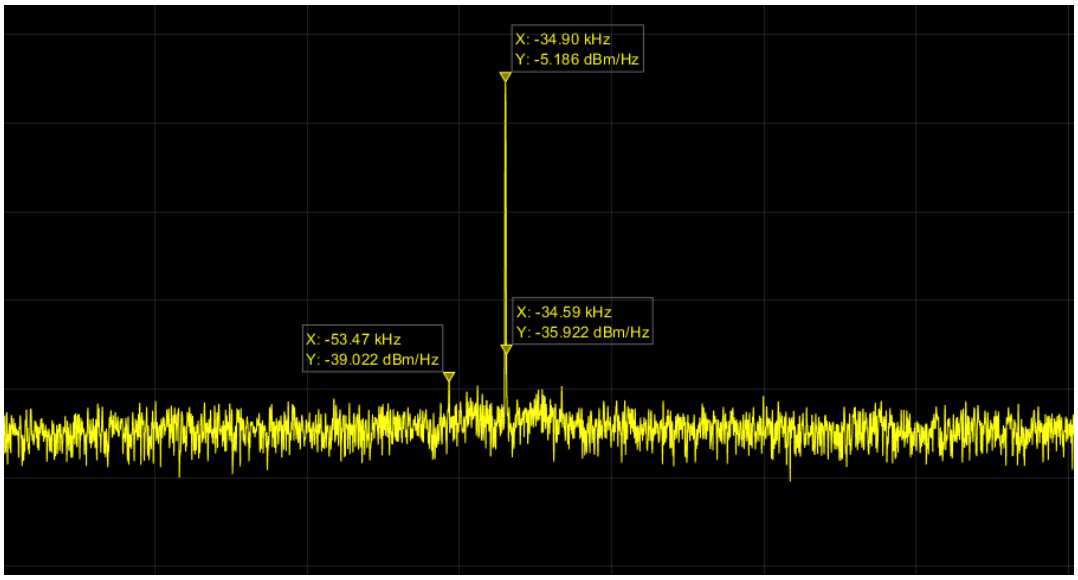


Figure 92. Initial spectrum analyzer frequency offset calibration.

This -4.515e4 Hz offset was added back to the center frequency of the receiver, shown in Figure 93 and Figure 94, and the calibration models were rerun to drive the offset to near zero as seen in Figure 95.

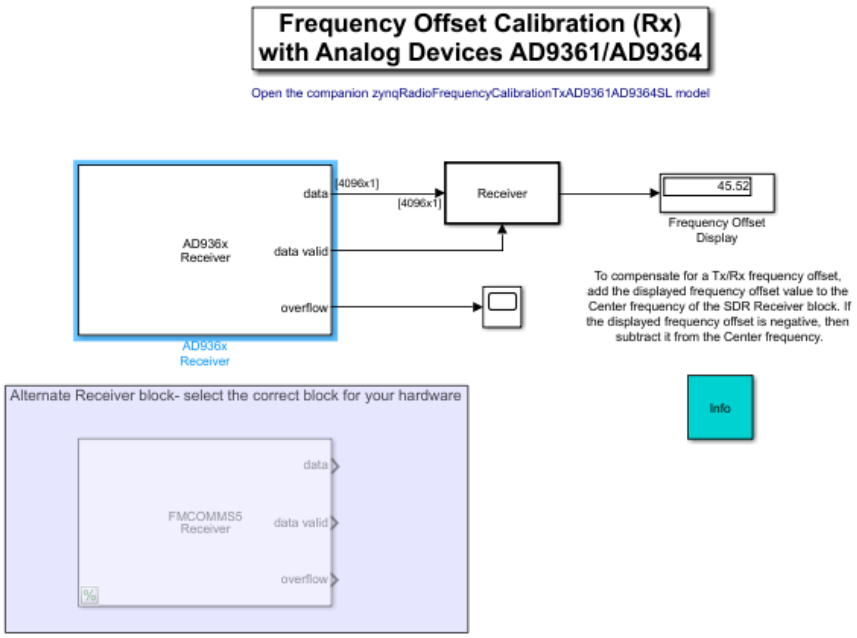


Figure 93. Tuned frequency offset calibration receiver model.

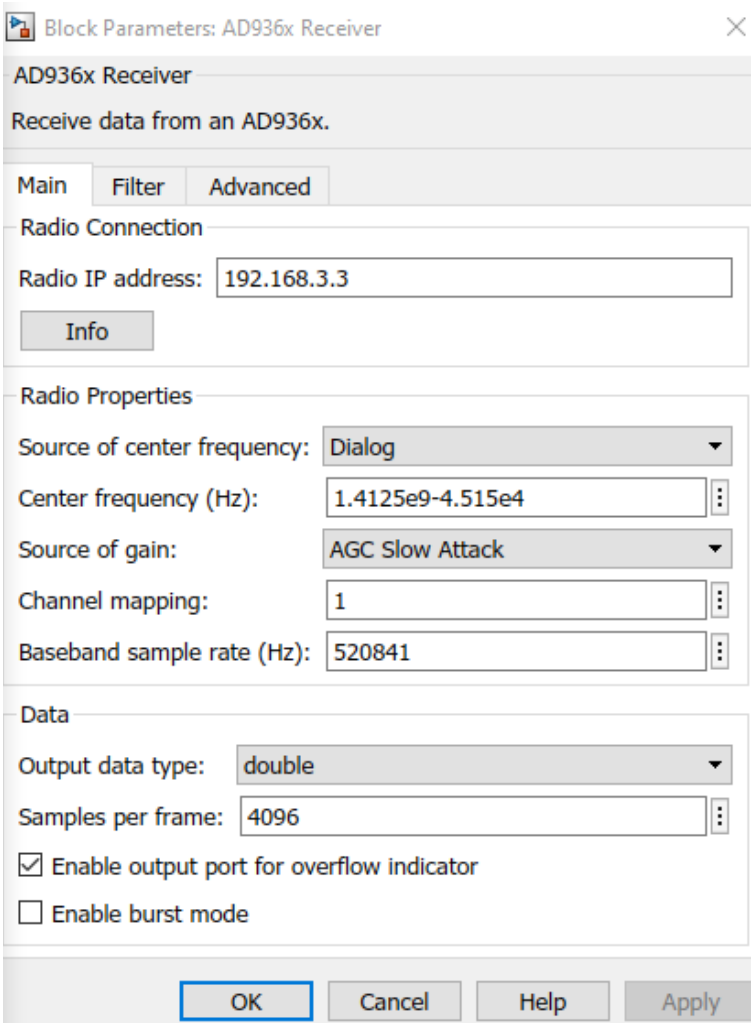


Figure 94. Tuned frequency offset calibration receiver block parameters.

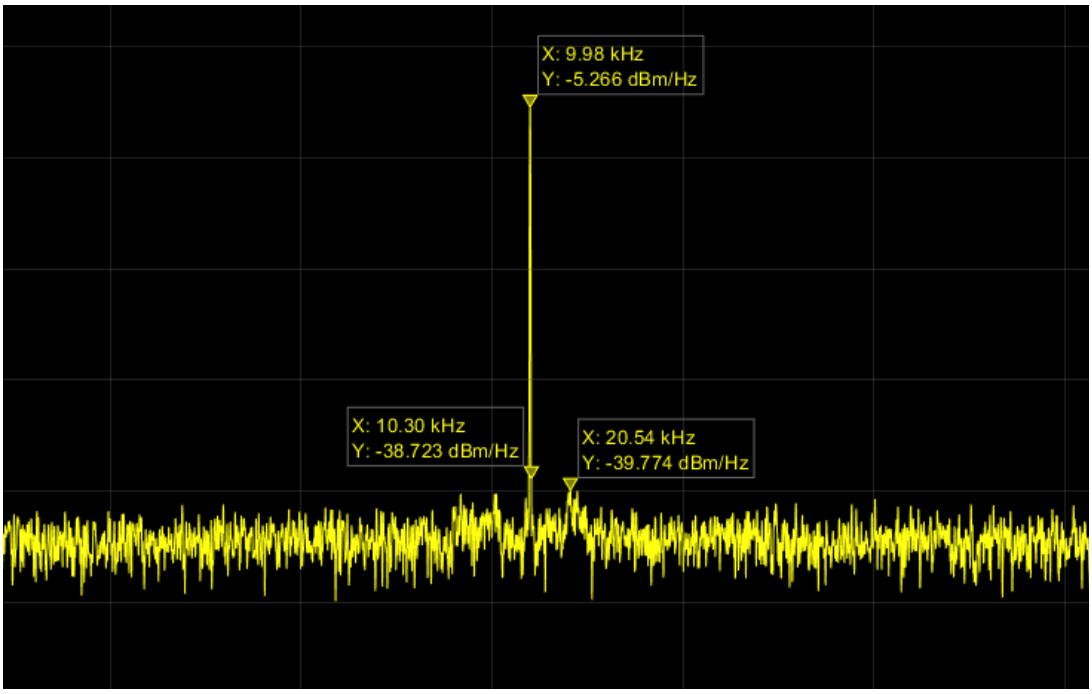


Figure 95. Tuned spectrum analyzer frequency offset calibration.

This same frequency offset was added to the center frequency of the QPSK receiver on the secondary PC. The QPSK transmitter model was executed first on the primary PC, followed by the calibrated QPSK receiver model on the secondary PC, with added frequency offset, as seen in Figure 96, Figure 97, Figure 98, and Figure 99.

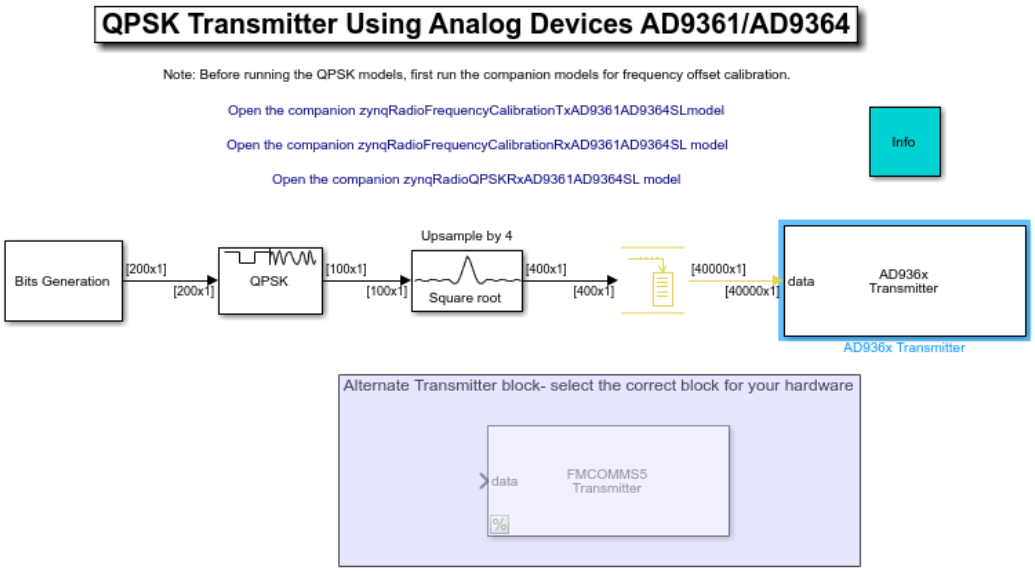


Figure 96. Functional testing QPSK transmitter model.

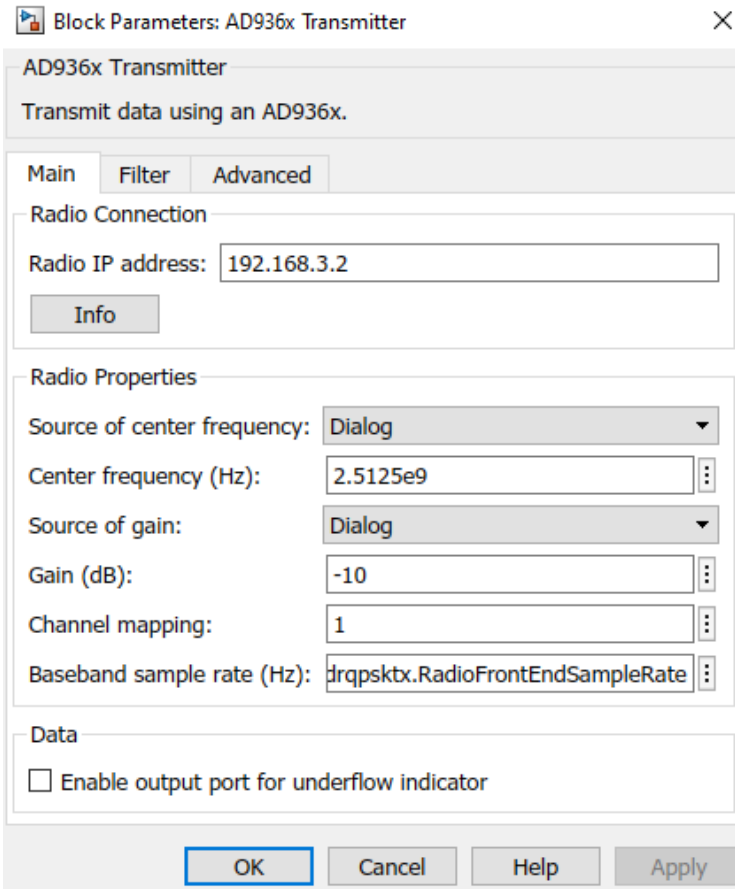


Figure 97. Functional testing QPSK transmitter block parameters.

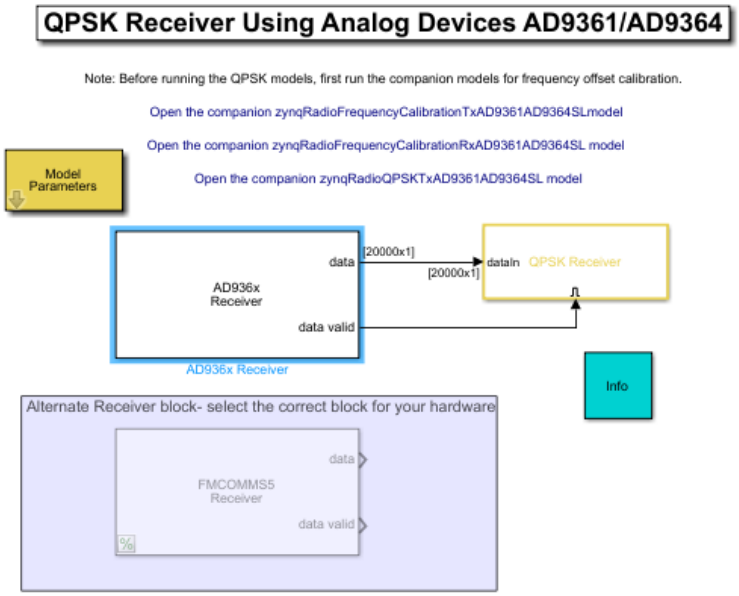


Figure 98. Functional testing QPSK receiver.

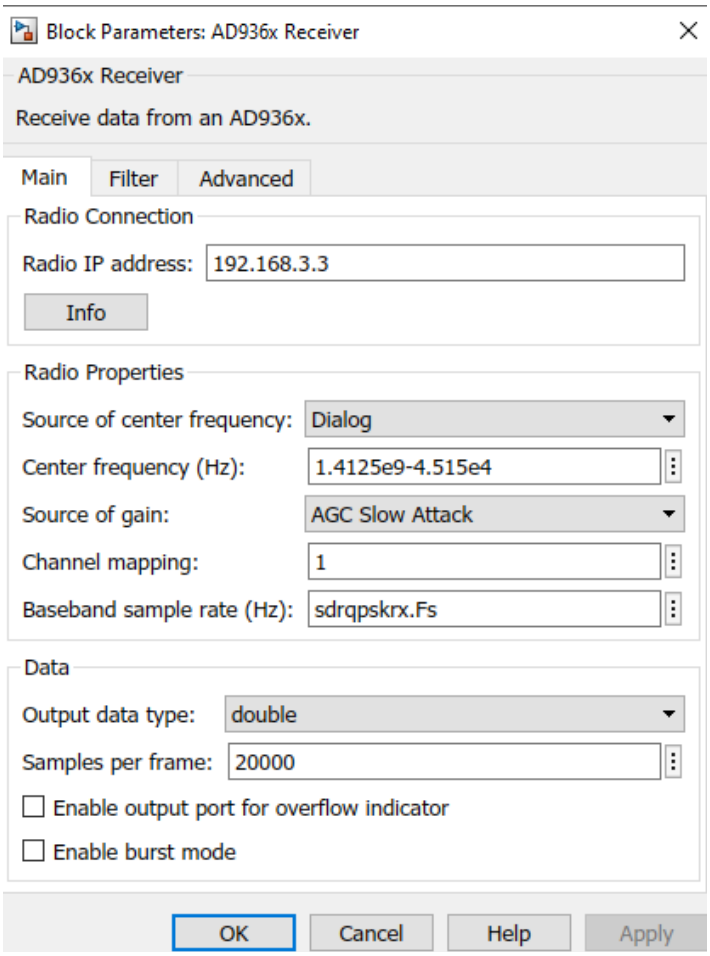


Figure 99. Functional testing QPSK receiver with frequency offset calibration block parameters.

Figure 100 shows the indexed “Hello world ###” decoded message source from the diagnostics viewer of the receiver model on the secondary PC, indicating a successful QPSK modulation/demodulation through the functional end-to-end bench testing of the X-band SDR system.

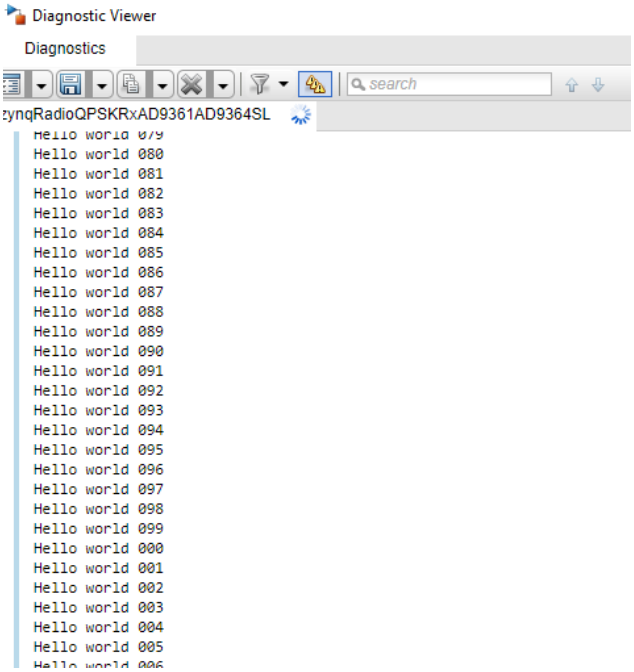


Figure 100. Functional testing successful QPSK modulation/demodulation.

Although transmission using QPSK modulation was successful, the default baseband sample rate limited the system data rate to 0.2604205 Mbps. To evaluate the correlation between the change in sample rate and the change in data rate, the sample period must be calculated, shown in Equation (7), where P_s is the sample period and F_s is the sample frequency in Hz. For this model, F_s = 520.841 kHz

$$P_s = \frac{1}{F_s}. \tag{7}$$

Frame time is then defined by Equation (8), where T_f is the frame time in seconds, S_f is the frame size and U is the up-sampling factor. For this model, S_f = 100 and U = 4

$$T_f = P_s \times S_f \times U. \tag{8}$$

Data rate is then found from Equation (9), where DR is data rate in bps and L_f is the frame length, which for this model is 200

$$DR = \frac{1}{\frac{1}{L_f} \times \left(T_f\right)}. \tag{9}$$

The calculated data rate for this test was 260420.5 bps, or 0.2604205 Mpbs. The channel characteristics of this default transmission are shown in Figure 101.

Figure 101. QPSK signal with 520841 Hz sample rate.

In analyzing the above equations, it is clear that increasing the baseband sample rate will in turn increase the data rate. The baseband sample rate is controlled in the QPSK transmit and receive models through the zynqRadioQPSKTxAD9361AD9364SL_init.m and zynqRadioQPSKRxAD9361AD9364SL_init.m initial parameters scripts (Appendix D and Appendix E, respectively). The parameter in these scripts which controls sample rate is “SimParams.RadioFrontEndSampleRate” and is defaulted to 520.841 kHz. The author iteratively increased this sample rate, and ran the QPSK transmit and receive models to determine the effective data rate, while still successfully modulating, demodulating, and decoding the message source sent by the transmitter to an indexed “Hello world ###”

message. At a sample rate above 4e6 Hz, associated with a data rate of 2 Mbps, the decoded message at the QPSK receiver began to fail recognizing symbols, as seen in Figure 102.

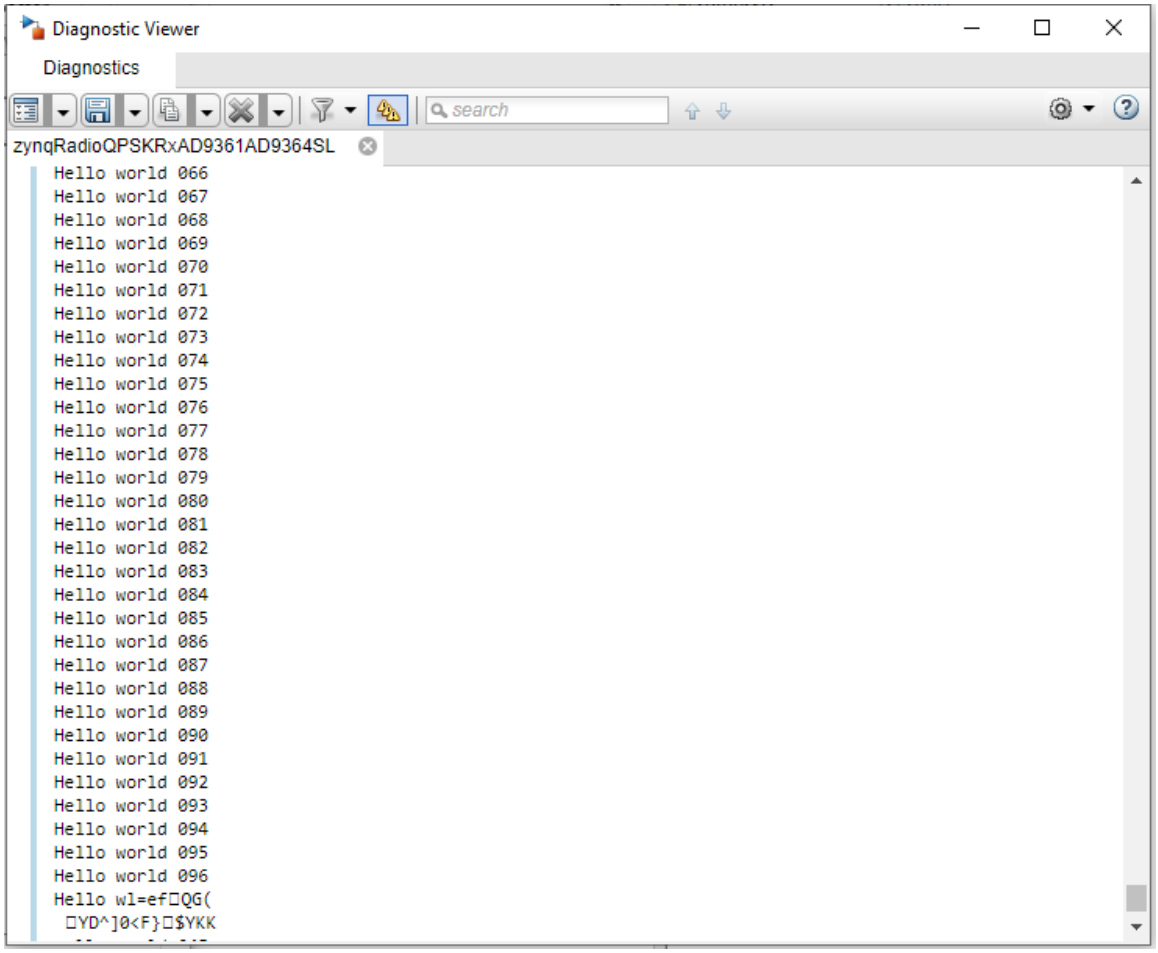


Figure 102. Decoding with errors at 4e6 Hz sample rate.

When the sample rate was increased past 4e6, decoding worsened and the receiver was unable to translate much of the message source, as shown in Figure 103.

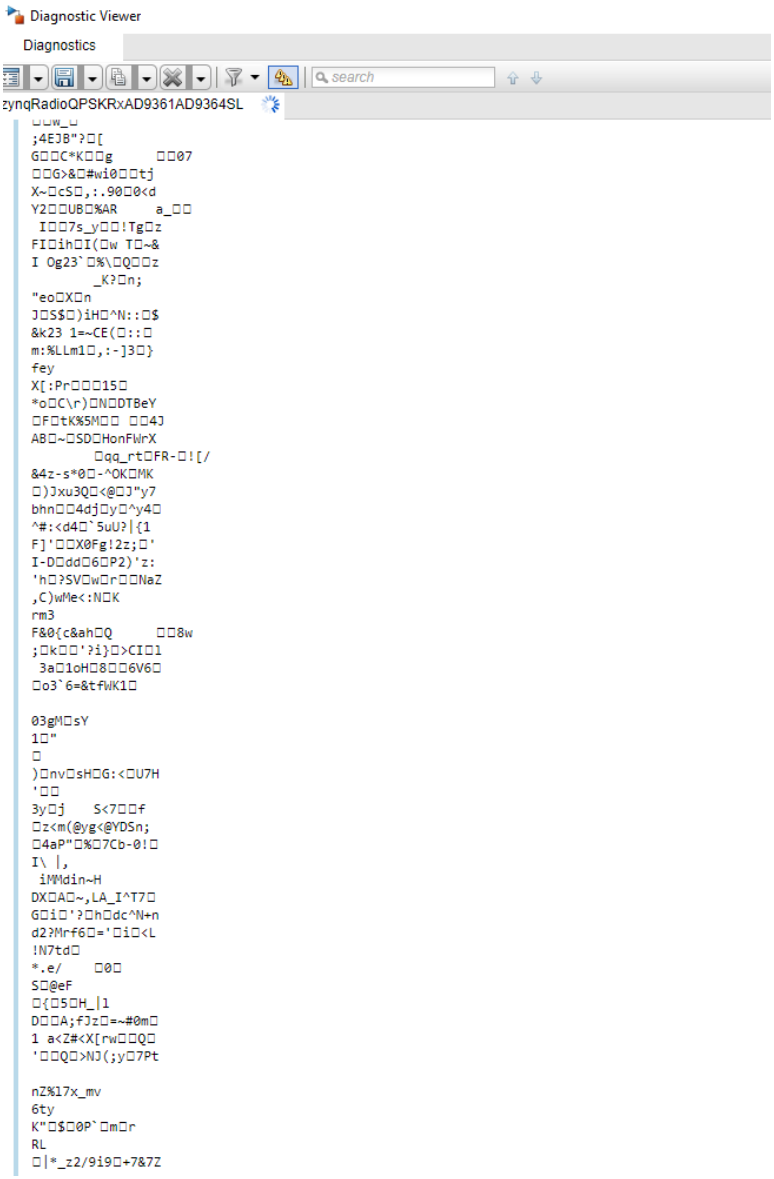


Figure 103. Decoding with errors at high sample rates.

Through this test, the author identified the that sample rate limit of the system was approximately 4e6 Hz, translating to a maximum data rate of 2 Mbps, after which decoding errors started to manifest. The output of this signal from the LNB was connected to the spectrum analyzer to evaluate characteristics. Screen captures from the spectrum analyzer are seen in Figure 104 and Figure 105.

Figure 104. Functional testing QPSK signal.

Figure 105. Functional testing QPSK signal occupied bandwidth.

Following the conclusion of functional end-to-end bench testing of the X-band SDR payload, the components of the payload were measured to ensure the mass requirement objective of 1.0 kg was met. The summary of these measurements is shown in Table 8.

Table 8. X-band SDR payload mass.

Component	**Mass (+/- 0.5g)**
High-power amplifier (HPA)	29.0
SoM	55.5
Carrier board	65.0
Convert board	6
Connectors	5
Transmission lines	5
Total mass	**165.5**

VI. CONCLUSIONS

A. SUMMARY

This research focused on the designing, building, and testing of a low-cost, high-bandwidth X-band SDR payload transmitter for CubeSat applications. Work conducted during this thesis demonstrated that an X-band SDR payload can be constructed from COTS components at a cost-effective rate, which is significantly cheaper than existing commercial alternatives. The incorporation of an SoM, drop-in RF components, and readily available software, utilizing QPSK modulation, allowed for reduced cost, size and weight into a form factor more easily integrated into CubeSats. Research requirements were driven by payload integration into the Corvus-6 bus within an approximately 0.5U radio assembly mechanical enclosure. Ethernet was selected as the data interface between the bus and X-band SDR payload. A data rate of 1 Mbps or greater using QPSK modulation was identified as a requirement to support the high data capacity of the TIC. COTS components, such as the ADRV9361-Z7035 SoM and the X-MW drop-in convert board, were chosen for this design iteration to provide nominal processing power and RF capability for CubeSat SDR applications, while reducing cost and shortening the development life cycle. A sufficient transmission data rate of 2 Mbps was effectively demonstrated during component-level, integrated, and functional end-to-end testing of the X-band SDR payload.

Following the design and construction of this payload, a three-phased laboratory testing approach was conducted to evaluate proper operation and measure performance of the hardware/software combination in relation to the research requirements. First, the individual components were tested to include the SoM with carrier board, X-MW convert board, and the X-band PLL LNB. Component-level testing verified an SoM data transmission rate over 1 Mbps, utilizing QPSK modulation, and confirmed the proper up-conversion and down-conversion of the X-MW convert board and LNB respectively. Integrated testing was then executed to combine components at a basic level in preparation for functional testing. This testing proved the convert board could successfully upconvert the QPSK-modulated signal output from the SoM. Additionally, it demonstrated that the

ZedBoard/ADFMComms3-EBZ, operated by a second PC, could successfully receive, demodulate, and decode transmissions from the SoM. Finally, functional end-to-end testing combined all components of the X-band SDR payload to test for transmission data rate using QPSK modulation and verify the size and weight requirements of 0.5U and 1 kg.

Lastly, the X-band SDR payload transmission data rate was limited to 2 Mbps, which was established during functional end-to-end testing. This was largely due to the use of SDR hardware as a peripheral controlled through a host PC rather than as a real-time operating system. Similarly, data transmission rate was hindered by the use of the SDR to perform all coding, interleaving, and randomization rather than dedicating these processes to be run by the SoC. While the primary input PC used to operate the SoM featured USB 3.0, the secondary PC used to receive the signal on the ZedBoard/ADFMComms-3 EBZ SDR was limited to USB 2.0. USB 2.0 is ten times slower than USB 3.0, and was likely a contributing factor which prevented the payload from achieving nominal data rate during transmission.

B. FUTURE WORK

1. Standalone System Using RTOS

Initial software research conducted during this thesis identified the use of a standalone RTOS for embedded processor applications as the preferred method for increasing SDR performance. Tools such as the MATLAB Embedded Coder, HDL Coder, Vivado Design Suite, and Vitis IDE, allow for MATLAB code to be deployed as a bitstream and implemented on the SoC. It is recommended that future design iterations implement embedded processor design directly on the SoC to increase transmission data rate capabilities. Transmission data rate was also limited during this research by the use of the ZedBoard/ADFMComms3-EBZ SDR operated through a second PC with USB 2.0 connectivity. Future iterations should implement USB 3.0 data interfaces to maximize effective transmission rates.

2. Simulink Model Development

This research utilized MATLAB Simulink example models for QPSK transmit and receive. While these models work well for initial testing, they do not allow the user to change all parameters or provide the ability to easily add or remove Simulink blocks. Future design iterations should develop custom QPSK models, referencing the models used by Bower [29], to maximize adaptability and configurability that better meets the needs of the MC3 network. It is also recommended that more intricate modulation schemes, such as OQPSK, are investigate for SDR applications. Similarly, future models should seek to incorporate Reed Solomon channel coding for FEC, interleaving, and randomization in accordance with the Consultative Committee for Space Data Systems (CCSDS) communication standards for spaceflight. These features were not included in the reference QPSK modulation models utilized during testing.

3. Integration with MC3 Network Receivers

The ZedBoard/ADFMComms3-EBZ SDR was used as a receiver during testing. This combination is intended for prototyping and development purposes rather than as a dedicated ground receiver. MC3 network receivers need to be identified with specific application for the X-band SDR payload to maximize interoperability and achieve nominal data transmission rates. Furthermore, functional end-to-end testing should be conducted along the full path from the X-band SDR payload to the dedicated MC3 network receiver.

4. Integration of RF Components into a PCB

Prototypes built using the X-MW design system approach are ready for production. Likewise, modular RF blocks can be rapidly integrated into a single PCB and offer standardized or custom housing to package a system and provide RF shielding. Production boards remove the need for anchors and gsgJumpers and provide grooved access for bias controller wiring. The development time from completed design to fully-tested product delivery is estimated to be from 12–14 weeks. The complete board assembly, including PCB and housing is estimated to be approximately $12,000. While housing will likely not be necessary for mounting in the radio assembly mechanical enclosure, the custom PCB will eliminate weak points between RF blocks presented by the solderless interconnect

method used during prototyping. Examples of combined drop-in blocks from X-MW are shown in Figure 106.

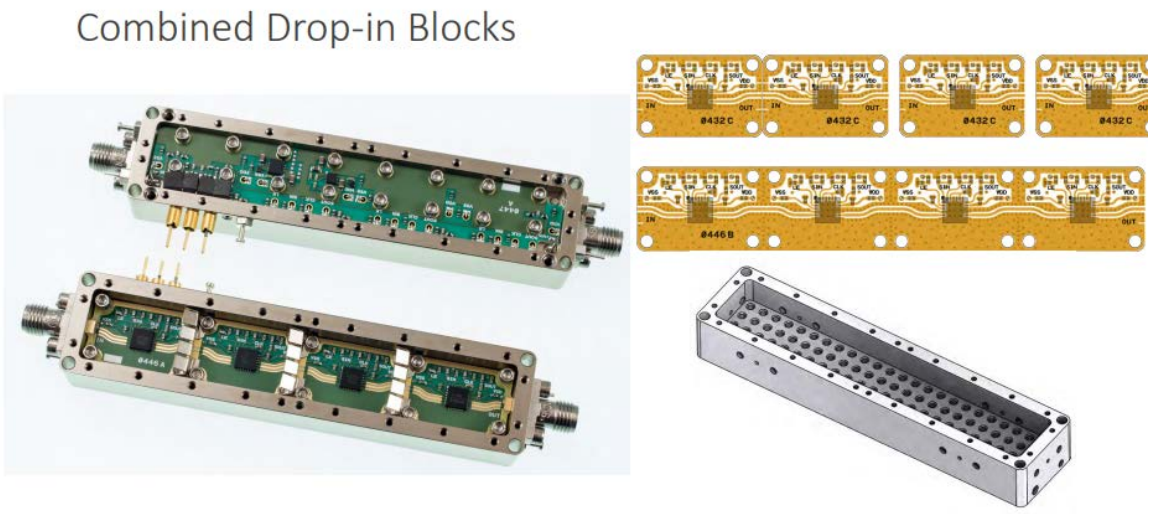


Figure 106. X-MW production example. Source: [38].

5. Environmental Testing

Functional testing was conducted on the X-band SDR payload to evaluate performance in relation to research requirements. However, environmental testing of the complete EDU in the radio assembly mechanical enclosure, with TVAC as a priority, needs to be conducted. This testing will evaluate the suitability of flight-hardware to survive launch and the expected LEO environment.

6. Radio Assembly Testing

During the course of this research, components were bench tested and were not fully assembled in the radio assembly mechanical enclosure. It is recommended that future work retest the X-band SDR payload, including the HPA, fully constructed inside the mechanical enclosure. This self-compatibility testing will evaluate that RF isolation is achieved to prevent RF leakage between components and ensure that the payload performs as expected.

APPENDIX A. LINK BUDGET SPREADSHEETS

Parameter	Symbol	Value	Units	Alt units				Mission Geometry					
Speed of light	c	300000000	m/s					R_earth	6378	km			
Frequency	f	8.21	Ghz					elevation angle	10	deg	0.174533	rad	
Tx power	P	4.00	W	6.021	dBW	36.021	dBm	altitude	1200	km			
Tx line loss	L_i	-1.00	dB					slant range (D)	3132.026	km			
Effective Tx antenna Gain	G_t	4.00	dB					Subtended angle	2.548	rad			
Equivalent Isotropically Radiated Power	EIRP	9.021	dBW					Earth Central angle	0.419	rad			
Slant range	d	3132.03	km										
Free space path loss	FSPL	-180.65	dB										
Propogation and polarization losses	L_a	-1.00	dB										
Theoretical System G/T	G/T	23.40	dB/K										
Data Rate	R	1200000	bps	60.79	dB-bps								
Implimentation loss	L_i	-1.50	dB						1 Mbps	->	1200 km		
Bit error rate	BER	0.00001	<N/A>						2.5 Mbps	->	1200 km		
Carrier power to noise power per unit bw	C/N0	77.87	dB-Hz						5 Mbps	->	725 km		
Achieved Eb/N0	Eb/N0	**17.08**	dB						10 Mbps	->	450 km		
Required Eb/N0	Req Eb/N0	9.65	dB										
Margin	LKM	**7.43**	dB										

Parameter	Symbol	Value	Units	Alt units				Mission Geometry					
Speed of light	c	300000000	m/s					R_earth	6378	km			
Frequency	f	8.21	Ghz					elevation angle	10	deg	0.174533	rad	
Tx power	P	4.00	W	6.021	dBW	36.021	dBm	altitude	500	km			
Tx line loss	L_i	-1.00	dB					slant range (D)	1695.081	km			
Effective Tx antenna Gain	G_t	4.00	dB					Subtended angle	2.722	rad			
Equivalent Isotropically Radiated Power	EIRP	9.021	dBW					Earth Central angle	0.245	rad			
Slant range	d	1695.08	km										
Free space path loss	FSPL	-175.32	dB										
Propogation and polarization losses	L_a	-1.00	dB										
Theoretical System G/T	G/T	23.40	dB/K										
Data Rate	R	10000000	bps	70.00	dB-bps								
Implimentation loss	L_i	-1.50	dB						1 Mbps	->	1200 km		
Bit error rate	BER	0.00001	<N/A>						2.5 Mbps	->	1200 km		
Carrier power to noise power per unit bw	C/N0	83.21	dB-Hz						5 Mbps	->	725 km		
Achieved Eb/N0	Eb/N0	**13.21**	dB						10 Mbps	->	450 km		
Required Eb/N0	Req Eb/N0	9.65	dB										
Margin	LKM	**3.56**	dB										

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. XILINX ZYNQ-7000 SOC DATA SHEET

The Xilinx Zynq-7000 SoC data sheet overview summarizes critical features of each SoC variant in the processor family [35].

XILINX

Zynq-7000 SoC Data Sheet: Overview

DS190 (v1.11.1) July 2, 2018 **Product Specification**

Zynq-7000 SoC First Generation Architecture

The Zynq®-7000 family is based on the Xilinx SoC architecture. These products integrate a feature-rich dual-core or single-core ARM® Cortex™-A9 based processing system (PS) and 28 nm Xilinx programmable logic (PL) in a single device. The ARM Cortex-A9 CPUs are the heart of the PS and also include on-chip memory, external memory interfaces, and a rich set of peripheral connectivity interfaces.

Processing System (PS)

ARM Cortex-A9 Based Application Processor Unit (APU)

- 2.5 DMIPS/MHz per CPU
- CPU frequency: Up to 1 GHz
- Coherent multiprocessor support
- ARMv7-A architecture
  - TrustZone® security
  - Thumb®-2 instruction set
- Jazelle® RCT execution Environment Architecture
- NEON™ media-processing engine
- Single and double precision Vector Floating Point Unit (VFPU)
- CoreSight™ and Program Trace Macrocell (PTM)
- Timer and Interrupts
  - Three watchdog timers
  - One global timer
  - Two triple-timer counters

Caches

- 32 KB Level 1 4-way set-associative instruction and data caches (independent for each CPU)
- 512 KB 8-way set-associative Level 2 cache (shared between the CPUs)
- Byte-parity support

On-Chip Memory

- On-chip boot ROM
- 256 KB on-chip RAM (OCM)
- Byte-parity support

External Memory Interfaces

- Multiprotocol dynamic memory controller
- 16-bit or 32-bit interfaces to DDR3, DDR3L, DDR2, or LPDDR2 memories
- ECC support in 16-bit mode
- 1GB of address space using single rank of 8-, 16-, or 32-bit-wide memories
- Static memory interfaces
  - 8-bit SRAM data bus with up to 64 MB support
  - Parallel NOR flash support
  - ONFI1.0 NAND flash support (1-bit ECC)
  - 1-bit SPI, 2-bit SPI, 4-bit SPI (quad-SPI), or two quad-SPI (8-bit) serial NOR flash

8-Channel DMA Controller

- Memory-to-memory, memory-to-peripheral, peripheral-to-memory, and scatter-gather transaction support

I/O Peripherals and Interfaces

- Two 10/100/1000 tri-speed Ethernet MAC peripherals with IEEE Std 802.3 and IEEE Std 1588 revision 2.0 support
  - Scatter-gather DMA capability
  - Recognition of 1588 rev. 2 PTP frames
  - GMII, RGMII, and SGMII interfaces
- Two USB 2.0 OTG peripherals, each supporting up to 12 Endpoints
  - USB 2.0 compliant device IP core
  - Supports on-the-go, high-speed, full-speed, and low-speed modes
  - Intel EHCI compliant USB host
  - 8-bit ULPI external PHY interface
- Two full CAN 2.0B compliant CAN bus interfaces
  - CAN 2.0-A and CAN 2.0-B and ISO 11898-1 standard compliant
  - External PHY interface
- Two SD/SDIO 2.0/MMC3.31 compliant controllers
- Two full-duplex SPI ports with three peripheral chip selects
- Two high-speed UARTs (up to 1 Mb/s)
- Two master and slave I2C interfaces
- GPIO with four 32-bit banks, of which up to 54 bits can be used with the PS I/O (one bank of 32b and one bank of 22b) and up to 64 bits (up to two banks of 32b) connected to the Programmable Logic
- Up to 54 flexible multiplexed I/O (MIO) for peripheral pin assignments

Interconnect

- High-bandwidth connectivity within PS and between PS and PL
- ARM AMBA® AXI based
- QoS support on critical masters for latency and bandwidth control

© Copyright 2012–2018 Xilinx, Inc., Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. AMBA, AMBA Designer, ARM, ARM Cortex-A9, CoreSight, Cortex, and PrimeCell are trademarks of ARM in the EU and other countries. PCI, PCIe, and PCI Express are trademarks of PCI-SIG and used under license. All other trademarks are the property of their respective owners.

Programmable Logic (PL)

Configurable Logic Blocks (CLB)

- Look-up tables (LUT)
- Flip-flops
- Cascadeable adders

36 Kb Block RAM

- True Dual-Port
- Up to 72 bits wide
- Configurable as dual 18 Kb block RAM

DSP Blocks

- 18 x 25 signed multiply
- 48-bit adder/accumulator
- 25-bit pre-adder

Programmable I/O Blocks

- Supports LVCMOS, LVDS, and SSTL
- 1.2V to 3.3V I/O
- Programmable I/O delay and SerDes

JTAG Boundary-Scan

- IEEE Std 1149.1 Compatible Test Interface

PCI Express® Block

- Supports Root complex and End Point configurations
- Supports up to Gen2 speeds
- Supports up to 8 lanes

Serial Transceivers

- Up to 16 receivers and transmitters
- Supports up to 12.5 Gb/s data rates

Two 12-Bit Analog-to-Digital Converters

- On-chip voltage and temperature sensing
- Up to 17 external differential input channels
- One million samples per second maximum conversion rate

Feature Summary

Table 1: **Zynq-7000 and Zynq-7000S SoCs**

	Device Name	Z-7007S	Z-7012S	Z-7014S	Z-7010	Z-7015	Z-7020	Z-7030	Z-7035	Z-7045	Z-7100
	Part Number	XC7Z007S	XC7Z012S	XC7Z014S	XC7Z010	XC7Z015	XC7Z020	XC7Z030	XC7Z035	XC7Z045	XC7Z100
Processing System	Processor Core	Single-core ARM Cortex-A9 MPCore™ with CoreSight™			Dual-core ARM Cortex-A9 MPCore™ with CoreSight™						
	Processor Extensions	NEON™ & Single / Double Precision Floating Point for each processor									
	Maximum Frequency	667 MHz (-1); 766 MHz (-2)			667 MHz (-1); 766 MHz (-2); 866 MHz (-3)			667 MHz (-1); 800 MHz (-2); 1 GHz (-3)			667 MHz (-1) 800 MHz (-2)
	L1 Cache	32 KB Instruction, 32 KB data per processor									
	L2 Cache	512 KB									
	On-Chip Memory	256 KB									
	External Memory Support(1)	DDR3, DDR3L, DDR2, LPDDR2									
	External Static Memory Support(1)	2x Quad-SPI, NAND, NOR									
	DMA Channels	8 (4 dedicated to Programmable Logic)									
	Peripherals(1)	2x UART, 2x CAN 2.0B, 2x I2C, 2x SPI, 4x 32b GPIO									
	Peripherals w/ built-in DMA(1)	2x USB 2.0 (OTG), 2x Tri-mode Gigabit Ethernet, 2x SD/SDIO									
	Security(2)	RSA Authentication, and AES and SHA 256-bit Decryption and Authentication for Secure Boot									
Processing System to Programmable Logic Interface Ports (Primary Interfaces & Interrupts Only)		2x AXI 32b Master 2x AXI 32-bit Slave 4x AXI 64-bit/32-bit Memory AXI 64-bit ACP 16 Interrupts									

Table 1: **Zynq-7000 and Zynq-7000S SoCs *(Cont'd)***

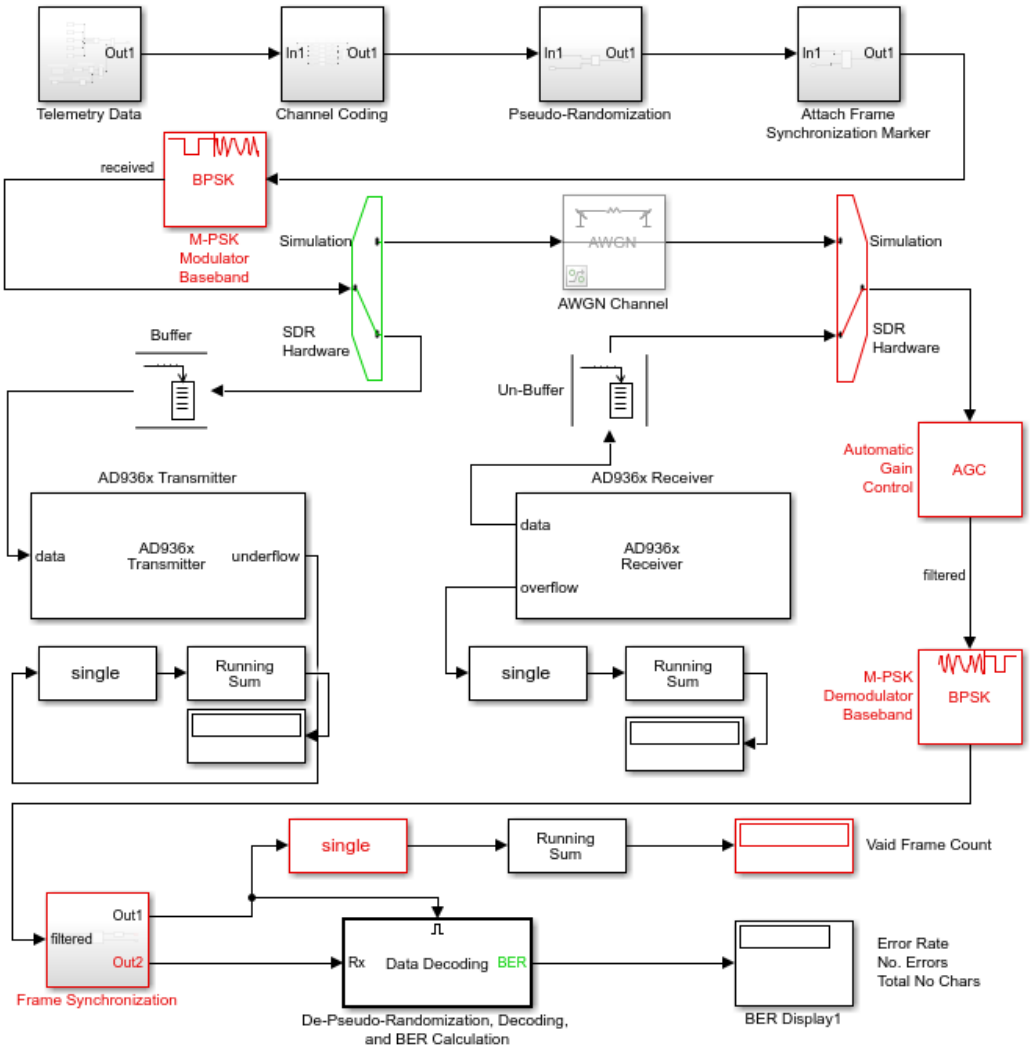
	Device Name	Z-7007S	Z-7012S	Z-7014S	Z-7010	Z-7015	Z-7020	Z-7030	Z-7035	Z-7045	Z-7100
	Part Number	XC7Z007S	XC7Z012S	XC7Z014S	XC7Z010	XC7Z015	XC7Z020	XC7Z030	XC7Z035	XC7Z045	XC7Z100
Programmable Logic	Xilinx 7 Series Programmable Logic Equivalent	Artix®-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Kintex®-7 FPGA	Kintex-7 FPGA	Kintex-7 FPGA	Kintex-7 FPGA
	Programmable Logic Cells	23K	55K	65K	28K	74K	85K	125K	275K	350K	444K
	Look-Up Tables (LUTs)	14,400	34,400	40,600	17,600	46,200	53,200	78,600	171,900	218,600	277,400
	Flip-Flops	28,800	68,800	81,200	35,200	92,400	106,400	157,200	343,800	437,200	554,800
	Block RAM (# 36 Kb Blocks)	1.8 Mb (50)	2.5 Mb (72)	3.8 Mb (107)	2.1 Mb (60)	3.3 Mb (95)	4.9 Mb (140)	9.3 Mb (265)	17.6 Mb (500)	19.2 Mb (545)	26.5 Mb (755)
	DSP Slices (18x25 MACCs)	66	120	170	80	160	220	400	900	900	2,020
	Peak DSP Performance (Symmetric FIR)	73 GMACs	131 GMACs	187 GMACs	100 GMACs	200 GMACs	276 GMACs	593 GMACs	1,334 GMACs	1,334 GMACs	2,622 GMACs
	PCI Express (Root Complex or Endpoint)[3]		Gen2 x4			Gen2 x4		Gen2 x4	Gen2 x8	Gen2 x8	Gen2 x8
	Analog Mixed Signal (AMS) / XADC	2x 12 bit, MSPS ADCs with up to 17 Differential Inputs									
	Security[2]	AES and SHA 256b for Boot Code and Programmable Logic Configuration, Decryption, and Authentication									

Notes:

1. Restrictions apply for CLG225 package. Refer to the UG585, *Zynq-7000 SoC Technical Reference Manual* (TRM) for details.
2. Security is shared by the Processing System and the Programmable Logic.
3. Refer to PG054, *7 Series FPGAs Integrated Block for PCI Express* for PCI Express support in specific devices.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. ORIGINAL SIMULINK SDR MODEL



THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D. ZYNQRADIOQPSKTXAD9361AD9364SL_INIT.M

```
function SimParams = zynqRadioQPSKTxAD9361AD9364SL_init
% Set simulation parameters



%SDR receiver parameters
SimParams.RadioFrontEndSampleRate = 520.841e3;
SimParams.RadioFrontEndSamplePeriod = 1 / SimParams.RadioFrontEndSampleRate;
SimParams.RadioChannelMapping = 1;

% General simulation parameters
SimParams.Upsampling = 4; % Upsampling factor
SimParams.Fs = SimParams.RadioFrontEndSampleRate; % Sample rate
SimParams.Ts = SimParams.RadioFrontEndSamplePeriod; % Sample time
SimParams.FrameSize = 100; % Number of modulated symbols per frame
SimParams.FrameTime = SimParams.Ts * SimParams.FrameSize * SimParams.Upsampling;
% Tx parameters
SimParams.BarkerLength = 13; % Number of Barker code symbols
SimParams.DataLength = (SimParams.FrameSize - SimParams.BarkerLength)*2; % Number of data
payload bits per frame
SimParams.MsgLength = 105; % Number of message bits per frame, 7 ASCII characters

SimParams.RxBufferedFrames = 10; % Received buffer length (in frames)
SimParams.RCFiltSpan = 10; % Group delay of Raised Cosine Tx Rx filters (in symbols)
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX E. ZYNQRADIOQPSKRXAD9361AD9364SL_INIT.M

```
function SimParams = zynqRadioQPSKRxAD9361AD9364SL_init
% Set simulation parameters
% SimParams = sdrzqpskrxFMC234_init



%SDR receiver parameters
SimParams.RadioCenterFrequency=2.4e9;
SimParams.RadioFrontEndSampleRate = 520.841e3;
SimParams.RadioFrontEndSamplePeriod = 1 / SimParams.RadioFrontEndSampleRate;

% General simulation parameters
SimParams.M = 4; % M-PSK alphabet size
SimParams.Upsampling = 4; % Upsampling factor
SimParams.Downsampling = 2; % Downsampling factor
SimParams.Fs = SimParams.RadioFrontEndSampleRate; % Sample rate
SimParams.Ts = SimParams.RadioFrontEndSamplePeriod; % Sample time
SimParams.FrameSize = 100; % Number of modulated symbols per frame

% Tx parameters
SimParams.BarkerLength = 13; % Number of Barker code symbols
SimParams.DataLength = (SimParams.FrameSize - SimParams.BarkerLength)*2; % Number of data
payload bits per frame
SimParams.MsgLength = 105;

% Rx parameters
SimParams.RxBufferedFrames = 10; % Received buffer length (in frames)
SimParams.RCFiltSpan = 10; % Filter span of Raised Cosine Tx Rx filters (in symbols)
SimParams.RadioFrameSize = SimParams.Upsampling * SimParams.FrameSize *
SimParams.RxBufferedFrames;
K = 1;
A = 1/sqrt(2);
% Look into model for details for details of PLL parameter choice.
SimParams.FineFreqPEDGain = 2*K*A^2+2*K*A^2; % K_p for Fine Frequency Compensation PLL,
determined by 2KA^2 (for binary PAM), QPSK could be treated as two individual binary PAM
SimParams.FineFreqCompensateGain = 1; % K_0 for Fine Frequency Compensation PLL
SimParams.TimingRecTEDGain = 2.7*2*K*A^2+2.7*2*K*A^2; % K_p for Timing Recovery PLL,
determined by 2KA^2*2.7 (for binary PAM), QPSK could be treated as two individual binary
PAM, 2.7 is for raised cosine filter with roll-off factor 0.5
SimParams.TimingRecCompensateGain = -1; % K_0 for Timing Recovery PLL, fixed due to
modulo-1 counter structure
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1] B. Cooper, *Astro Digital Corvus-6 spacecraft platform*, 2015.

[2] A. Johnstone, "CubeSat design specification (1U-12U) rev 14 CP-CDS-R14." The CubeSat Program, Cal Poly SLO, July 2020. [Online]. Available: https://static1.squarespace.com/static/5418c831e4b0fa4ecac1bacd/t/5f24997b6deea10cc52bb016/1596234122437/CDS+REV14+2020-07-31+DRAFT.pdf.

[3] National Aeronautics and Space Administration, "CubeSat 101: Basic concepts and processes for first-time CubeSat developers," October 2017. [Online]. Available: https://www.nasa.gov/sites/default/files/atoms/files/nasa_csli_cubesat_101_508.pdf.

[4] E. Mabrouk, "What are SmallSats and CubeSats?," NASA, March 13, 2015. [Online]. Available: http://www.nasa.gov/content/what-are-smallsats-and-cubesats.

[5] A. Heiney, "CubeSat launch initiative introduction," NASA, October 27, 2020. [Online]. Available: http://www.nasa.gov/content/cubesat-launch-initiative-introduction.

[6] Jet Propulsion Laboratory, "JPL CubeSat MarCO," 2020. [Online]. Available: https://www.jpl.nasa.gov/cubesat/missions/marco.php.

[7] G. Budris, "The Atlas V Aft Bulkhead Carrier requirements for the small satellite designer," *24th Annual AIAA/USU Conf. on Small Satellites*, 2010. [Online]. Available: https://www.ulalaunch.com/docs/default-source/rideshare/the-atlas-v-aft-bulkhead-carrier---requirements-for-the-small-satellite-designer-(aiaa-usu-small-sat-2010).pdf.

[8] G. Minelli, L. Magallanes, N. Weitz, D. Rigmaiden, J. Horning, and J. Newman, "The Mobile CubeSat Command and Control (MC3) ground station network: an overview and look ahead," *33rd Annual AIAA/USU Conf. on Small Satellites*, 2019. [Online]. Available: https://digitalcommons.usu.edu/smallsat/2019/all2019/85/.

[9] S. H. Wood, "Trade study of commercial software-defined radio technologies for small satellite ground station network command and control applications," M.S. thesis, Space Sys. Academic Group, NPS, Monterey, CA, USA, 2020. [Online]. Available: https://calhoun.nps.edu/handle/10945/65475.

[10] Federal Communications Commission, “Radio spectrum allocation,” March 02, 2011. [Online]. Available: https://www.fcc.gov/engineering-technology/policy-and-rules-division/general/radio-spectrum-allocation.

[11] FCC Office of Engineering and Technology Policy and Rules Division, “FCC online table of frequency allocations,” October 09, 2020. [Online]. Available: https://transition.fcc.gov/oet/spectrum/table/fcctable.pdf.

[12] International Telecommunication Union, “About ITU,” 2020. [Online]. Available: https://www.itu.int/en/about/Pages/default.aspx.

[13] Innoflight, *Innoflight Compact (L+S)/S Radio (SCR-104)*, 2020. [Online]. Available: https://www.innoflight.com/product-overview/scrs/scr-104/.

[14] Innoflight, *Innoflight Compact X-Band Transceiver (SCR-106)*, 2020. [Online]. Available: https://www.innoflight.com/product-overview/scrs/scr-106/.

[15] S. J. Kline, “Flight qualification of a terahertz imaging camera as a CubeSat payload,” M.S. thesis, Space Sys. Academic Group, NPS, Monterey, CA, USA, 2018. [Online]. Available: https://calhoun.nps.edu/handle/10945/59702.

[16] B. Sklar, *Digital Communications: Fundamentals and Applications*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2001.

[17] M. Kolawole, *Satellite Communication Engineering*, 2nd ed. Boca Raton, FL, USA: CRC Press, Taylor & Francis Group, LLC, 2013.

[18] T. Kuphaldt, *Understanding I/Q Signals and Quadrature Modulation Radio Frequency Demodulation Electronics Textbook*, 2000. [Online]. https://www.allaboutcircuits.com/textbook/radio-frequency-analysis-design/radio-frequency-demodulation/understanding-i-q-signals-and-quadrature-modulation/.

[19] R. Stewart, K. Barlee, D. Atkinson, and L. Crockett, *Software Defined Radio using MATLAB & Simulink and the RTL-SDR*, 1st ed. Glasgow, Scotland, UK: Strathclyde Academic Media, 2015.

[20] W. Tuttlebee, *Software Defined Radio: Enabling Technologies*, 2nd ed. New York, NYm USA: J. Wiley & Sons, 2002.

[21] D. Oikonomou, P. Nomikos, G. Limnaios, and K. Zikidis, “Passive radars and their use in the modern battlefield,” *Journal of Computations & Modeling*, vol. 9, no.2, pp. 37–61, 2019.

[22] IQ Spacecom, *XLink X Band Transceiver SDR for Small Satellites*, June 2019. [Online]. Available: https://www.iq-spacecom.com/images/downloads/XLink_Datasheet_06.2019.pdf.

[23] GomSpace, *GOMspace NanoCom XT8250*, 2020. [Online]. Available: https://gomspace.com/shop/subsystems/communication-systems/nanocom-xt8250.aspx.

[24] J. Wertz, D. Everett, and J. Puschell, *Space Mission Engineering: The New SMAD*, Hawthorne, CA, USA: Microcosm Press, 2011.

[25] SS4861 NPS Payload Design Course, "X-Band SDR interface control document." March 17, 2020.

[26] B. L. Lovdahl, "Software-defined radio payload design for CubeSat and X-band communications," M.S. thesis, Space Sys. Academic Group, NPS, Monterey, CA, USA, 2018. [Online]. Available: https://calhoun.nps.edu/handle/10945/61218.

[27] G. Bischoff, "X-band software-defined radio for CubeSat applications," M.S. thesis, Space Sys. Academic Group, NPS, Monterey, CA, USA, 2019. [Online]. Available: https://calhoun.nps.edu/handle/10945/63431.

[28] AVNET, "ZedBoard SDR II evaluation kit," 2017. [Online]. Available: http://zedboard.org/product/zedboard-sdr-ii-evaluation-kit.

[29] E. Bower, "Software defined radio implementation using Mathworks Simulink and MATLAB with commercially available hardware," NPS, Monterey, CA, USA, 2020.

[30] AVNET, *ZedBoard Getting Started Guide version 7.0*, 2014. [Online]. Available: http://zedboard.org/sites/default/files/documentations/GS-AES-Z7EV-7Z020-G-V7-1.pdf.

[31] AVNET, *ZedBoard Hardware User's Guide version 2.2*, 2014. [Online]. Available: http://zedboard.org/sites/default/files/documentations/ZedBoard_HW_UG_v2_2.pdf.

[32] Xilinx, *Zynq-7000 SoC Technical Reference Manual*, 2018. [Online]. Available: https://www.xilinx.com/support/documentation/user_guides/ug585-Zynq-7000-TRM.pdf.

[33] Xilinx, *Zynq-7000 SoC Product Selection Guide*, 2014. [Online]. Available: https://www.xilinx.com/support/documentation/selection-guides/zynq-7000-product-selection-guide.pdf.

[34] P. Di, A. Cozma, and T. Hill, "Four quick steps to production: using model-based design for software-defined radio. Part 1 - the Analog Devices/Xilinx SDR rapid prototyping platform: Its capabilities, benefits, and tools," Sep. 2015. [Online]. Available: https://www.analog.com/en/analog-dialogue/articles/using-model-based-design-sdr-1.html.

[35] Xilinx, *Zynq-7000 SoC Data Sheet: Overview*, 2018. [Online]. Available: https://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf.

[36] Analog Devices, *ADFMComms3-EBZ User Guide*, 2020. [Online]. Available: https://wiki.analog.com/resources/eval/user-guides/ad-fmcomms3-ebz.

[37] Analog Devices, *AD9361 RF Agile Transceiver datasheet (Rev. F)*, 2016. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/AD9361.pdf.

[38] E. Luther, "RF & microwave system design: transforming the industry with modular building blocks," Austin, TX, 2019. [Online]. Available: https://www.xmicrowave.com/library/presentations/.

[39] X-Microwave, *XM-A9W8-0404C-01 X-MWblock*, 2020. [Online]. Available: https://www.xmicrowave.com/product/xm-a9w8-0404c-01/.

[40] X-Microwave, *X-Microwave modular design, prototyping, and production from DC to 95 GHz, 2019*. [Online]. Available: https://www.xmicrowave.com/library/brochures/.

[41] X-Microwave, *XM-GSGJ-01*, 2020. [Online]. Available: https://www.xmicrowave.com/product/xm-gsgj-01/.

[42] X-Microwave, *X-Microwave Prototype Station, X-MWprotostation*, 2019. [Online]. Available: https://www.xmicrowave.com/library/brochures/.

[43] X-Microwave, *XM-PP2-3232-01*, 2020. [Online]. Available: https://www.xmicrowave.com/product/xm-pp2-3232-01/.

[44] Analog Devices, *Analog Devices ADF4356 Data Sheet Rev. A*, 2017. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ADF4356.pdf.

[45] X-Microwave, *Getting Started Guide: X-MWcontroller*. 2018. [Online]. Available: https://www.xmicrowave.com/wp-content/uploads/Getting-Started-Guide-X-MWcontroller_V2018_4_11EL.pdf.

[46] Analog Devices, *HMC1121 data sheet rev. A*, 2016. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/HMC1121.pdf.

[47] AVNET, *PicoZed SDR 2x2 product brief*, 2019. [Online]. Available: http://zedboard.org/sites/default/files/product_briefs/PB-AES-Z7PZ-SDR2-PicoZed-SDR-SOM-V1.pdf.

[48] Analog Devices, *ADRV9361-Z7035 System on Module (SoM) SDR User Guide*, 2020. [Online]. Available: https://wiki.analog.com/resources/eval/user-guides/adrv9361-z7035.

[49] AVNET, *PicoZed SDR breakout carrier product brief*, 2017. [Online]. Available: http://zedboard.org/sites/default/files/product_briefs/PB-AES-PZSDRCC-BRK-G-V1.pdf.

[50] AVNET, *PicoZed SDR Breakout Carrier Card Hardware User Guide Version 1.2*, 2016. [Online]. Available: http://zedboard.org/sites/default/files/documentations/AES-PZSDRCC-BRK-G_UG_v1.2.pdf.

[51] AVNET, *PicoZed SDR breakout carrier quick start card*, 2016. [Online]. Available: http://zedboard.org/sites/default/files/documentations/AES-PZSDRCC-BRK-G_Quick_Start_Card-V2.pdf.

[52] Rosenberger, *32K10A-40ML5 technical data sheet*, 2015. [Online]. Available: https://products.rosenberger.com/_ocassets/db/32K10A-40ML5.pdf.

[53] X-Microwave, *XM-ANCHOR2*, 2020. [Online]. Available: https://www.xmicrowave.com/product/xm-anchor2/ (accessed Sep. 05, 2020).

[54] Texas Instruments, *LP38798 LDO data sheet*, 2017. [Online]. Available: https://www.ti.com/lit/ds/symlink/lp38798.pdf.

[55] Qorvo, *CMD264P3 product data sheet Rev 1.1*, 2018. [Online]. Available: https://www.qorvo.com/products/p/CMD264P3#documents.

[56] Connor Winfield, *TB602-050.0M oscillator data sheet rev.10*, 2018. [Online]. Available: http://www.conwin.com/datasheets/tx/tx380.pdf.

[57] Analog Devices, *ADP7142 LDO Data Sheet Rev.H*, 2020. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ADP7142.pdf.

[58] Mini-Circuits, *HFCN-7150+ high pass filter data sheet*, 2015. [Online]. Available: https://www.minicircuits.com/pdfs/HFCN-7150+.pdf.

[59] Mini-Circuits, *BFCN-5750+ bandpass filter data sheet*, 2019. [Online]. Available: https://www.minicircuits.com/pdfs/BFCN-5750+.pdf.

[60] Analog Devices, *LTC5553 – 3GHz to 20GHz microwave mixer data sheet*, 2016. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/5553f.pdf.

[61] Analog Devices, *LT3045 LDO data sheet*, 2015. [Online]. Available: https://www.mouser.com/datasheet/2/609/LT3045-1667614.pdf.

[62] Mini-Circuits, *BFCN-2555+ bandpass filter data sheet*, 2016. [Online]. Available: https://www.minicircuits.com/pdfs/BFCN-2555+.pdf.

[63] X-Microwave, "X-Microwave cascade and layout tool." Accessed August 05, 2020. [Online]. Available: https://cascade.xmicrowave.com/.

[64] AVNET, *ZedBoard Configuration and Booting Guide v1.1*, 2012. [Online]. Available: https://forums.xilinx.com/xlnx/attachments/xlnx/ELINUX/8461/1/ZedBoard_boot_guide_IDS14_1_v1_1.pdf.

[65] MATLAB, "Frequency offset calibration using Analog Devices AD9361/AD9364 - MATLAB & Simulink example." https://www.mathworks.com/help/supportpkg/xilinxzynqbasedradio/ug/frequency-offset-calibration-using-analog-devices-ad9361-ad9364.html (accessed Oct. 08, 2020).

[66] MATLAB, "QPSK transmitter using Analog Devices AD9361/AD9364 - MATLAB & Simulink example." Accessed October 08, 2020. [Online]. Available: https://www.mathworks.com/help/supportpkg/xilinxzynqbasedradio/ug/qpsk-transmitter-using-analog-devices-ad9361-ad9364.html.

[67] MATLAB, "QPSK receiver using Analog Devices AD9361/AD9364 - MATLAB & Simulink example." Accessed October 08, 2020. [Online]. Available: https://www.mathworks.com/help/supportpkg/xilinxzynqbasedradio/ug/qpsk-receiver-using-analog-devices-ad9361-ad9364.html.

[68] Swedish Microwave, "X-band PLL LNB for DVB-S2X services." Accessed October 09, 2020. [Online]. Available: https://smw.se/product/rf/lnb/x-band-pll-lnb/.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California